


Good Practice Guide on Vulnerability Disclosure

From challenges to recommendations

NOVEMBER 2015

About ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of network and information security expertise for the European Union (EU), its member states, the private sector and Europe's citizens. ENISA works with these groups to develop advice and recommendations on good practice in information security. It assists EU member states in implementing relevant EU legislation and works to improve the resilience of Europe's critical information infrastructure and networks. ENISA seeks to enhance existing expertise in EU member states by supporting the development of cross-border communities committed to improving network and information security throughout the EU. More information about ENISA and its work can be found at www.enisa.europa.eu.

Authors

This document was created by the CERT Capability team at ENISA in consultation with RAND Europe

Project officer

Cosmin Ciobanu

Contact

To contact the authors please use cert-relations@enisa.europa.eu
For media enquiries about this report, please use press@enisa.europa.eu.

Acknowledgements

The project team wishes to thank all the interviewees, and those who provided written input for the project, for their time and invaluable insights. Their contribution to this project has been essential.

Legal notice

Notice must be taken that this publication represents the views and interpretations of the authors and editors, unless stated otherwise. This publication should not be construed to be a legal action of ENISA or the ENISA bodies unless adopted pursuant to the Regulation (EU) No 526/2013. This publication does not necessarily represent state-of the-art and ENISA may update it from time to time.

Third-party sources are quoted as appropriate. ENISA is not responsible for the content of the external sources including external websites referenced in this publication.

This publication is intended for information purposes only. It must be accessible free of charge. Neither ENISA nor any person acting on its behalf is responsible for the use that might be made of the information contained in this publication.

ISBN 978-92-9204-148-9, DOI 10.2824/610384, Catalogue Number: TP-01-15-893-EN-N

Table of Contents

Executive summary

Vulnerabilities are ‘flaws’ or ‘mistakes’ in computer-based systems that may be exploited to compromise the network and information security of affected systems. They provide a point-of-entry or gateway to exploit a system and as such pose potentially severe security risks. Identifying and fixing vulnerabilities is therefore crucial, and the process of disclosing vulnerabilities is a vital component that cannot be underestimated. The vulnerability disclosure landscape is complex, with several stakeholders involved that include vendors, IT security providers, independent researchers, the media, malicious users, governments and, ultimately, the general public. These stakeholders often have competing interests, which results in a challenging landscape.

In the specific context of the vulnerability disclosure process, this study seeks to achieve the primary objectives listed in the following figure (Figure 1).

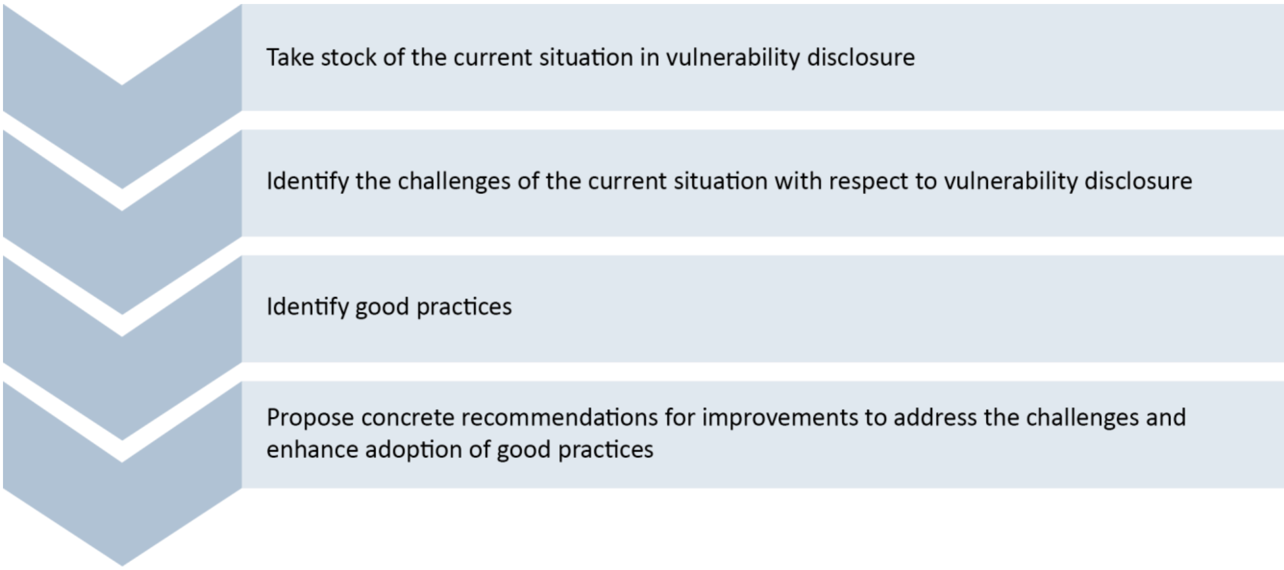


Figure 1: Primary objectives of the research

We used a mixed-method approach to accomplish the objectives outlined above, which included a focused literature review, a series of in-depth interviews with key experts in the field, and case studies of four widely distributed critical vulnerabilities reported in 2014.

Our analysis confirmed that there are a number of pressing **challenges** associated with the vulnerability disclosure process. The key challenges we identified are summarised below.

- ***Legal challenges:*** Individuals who discover a vulnerability often face legal threats when they decide to report it. These threats can have implications on not only civil and criminal law but also contract law, licensing, patent law and other types of legislation. Discoverers may find themselves in a grey area due to the methods used to discover the vulnerability and the way it was disclosed.
- ***Lack of ‘vendor’ maturity:*** Whereas large companies familiar within the information technology environment have robust processes in place for vulnerability reporting, other companies are new to the scene. These may be small companies, or companies which have not previously been part of the information technology landscape. This could lead to a lack of maturity on the part of these players and

potentially complicate the vulnerability disclosure eco-system, as less mature companies are ill-prepared to accept vulnerability reports and act upon them in the interest of the information security community.

- ***Lack of researcher maturity:*** Just as vendors may lack experience in accepting vulnerability reports, so some researchers lack experience in reporting vulnerabilities. When such experience is absent, researchers may approach vendors in a threatening or otherwise non-conducive manner which prevents fruitful cooperation. Researchers who lack experience may also be unwilling to compromise on, for example, timelines identified by the vendor.
- ***Incoming vulnerability reports are not always taken into consideration by the vendors.*** For all sorts of reasons vendors may choose to disregard reports about a vulnerability on their products and services in spite of the obvious damage they might suffer from not acting upon validated vulnerability information. Vulnerabilities reported may be singled out as academic or theoretical and interest for a previously disregarded vulnerability report might increase only after a security incident has happened when as it often happens, an ex post analysis is likely to set the record straight.
- ***Vulnerability acquisition for national intelligence purposes:*** Unpatched vulnerabilities can be used by criminals but also potentially by national intelligence or law enforcement officials. This means that sometimes a vulnerability will remain undisclosed for such (national) security purposes. Yet while a vulnerability remains undisclosed to the vendor, so will the development of a solution remain absent, leaving users vulnerable.
- ***Users do not implement patches (in a timely manner):*** Once a vulnerability is disclosed by the vendor, and a solution such as a patch to be installed via an update is available, the user must implement it. Such implementation is essential for the vulnerability to be resolved; a lack of implementation leaves users even more vulnerable since information about the vulnerability is now public. Users have a tendency, for a variety of reasons, to postpone or to negate patching altogether. This may be because of a lack of understanding or knowledge. Furthermore, it could be important to keep the average patch application time as short as possible rather than solely focusing on the disclosure timeline.
- ***Discoverer motivation varies:*** The motivation for an individual to discover a vulnerability varies. The motivation of the discoverer can influence the decision s/he makes regarding what to do with the vulnerability. The increase in bug bounty programmes as well as the growing zero-day market have increasingly placed a focus on monetary reward. This may lead to over-incentivising the search for vulnerabilities and may also lead to the expectation that discoverers will always receive a monetary reward for their discovery.

We also identified a number of **good practices** for the various stakeholders involved and these have been summarised below.

- ***Use existing documents:*** Previous efforts have been made to gather good practices in the area of vulnerability disclosure as well as to describe how to set up a vulnerability disclosure policy. To prevent 'reinvention of the wheel', these documents should be used by stakeholders and should be leveraged more by new initiatives. The ISO standards are a prominent example, although their lack of free availability may hamper their reach.
- ***Communication:***
  - **Vendors should be reachable/have a point of contact:** To prevent reporters from having to spend valuable time and resources looking for the appropriate contact, vendors should have a clear point of contact to deal with vulnerability reports, and this contact should be reachable.
  - **Have a policy in place:** Vendors should have a policy in place which addresses vulnerability disclosure and describes how they respond to vulnerability reports. This policy will also indicate to reporters what information they need to provide as well as give an insight into the process of the disclosure.

- **Communication with different stakeholders:** Communication with stakeholders requires mutual respect, patience and transparency. Continual communication is essential to acknowledge receipt of the vulnerability report as well as provide an indication of the next steps.

- ***Information dissemination:*** Information about the vulnerability as well as its solution, if available, should be disseminated to inform users of the developments and to provide them with an opportunity to protect themselves. How much information needs to be disseminated is a topic of discussion among stakeholders.
- ***Timelines:*** a timeline should be mutually agreed upon (on a vulnerability-by-vulnerability basis) to ensure that a vulnerability will be sufficiently addressed by the vendor in a timely manner.
- ***Flexibility:*** No 'one size fits all' approach exists in the area of vulnerability disclosure, so flexibility is necessary to tailor the vulnerability report as well as the response to the specifics of the vulnerability.

Finally, the synthesis of the various sources reviewed during this study shows that there are several areas for future consideration. Below, we summarise a series of **recommendations** for improving the status quo in the vulnerability disclosure landscape.

- ***The community must facilitate the improvement of vendor maturity:*** To make progress, vendor maturity must be improved to ensure that all vendors are able to receive vulnerability reports and respond to them in a manner which is accepted by the community and which will introduce the smallest risks with respect to the security of users. In this context the term community refers to different relevant stakeholders like EU Member States, vendors, security researchers, national CSIRTs and ENISA.
  In order to improve vendor maturity, the community must stimulate less mature vendors to introduce a policy and set up an infrastructure which allows them to accept vulnerability reports.
- ***Internationalisation through policy learning:*** The global nature of the internet requires a more transnational approach to the topic of vulnerability disclosure, where successful cases in certain countries or regions can be used for policy learning purposes in other areas of the world. Simultaneously, stakeholder gatherings at the transnational level can use their international access to further enhance such policy learning, and so allow the spread of good practices in vulnerability disclosure.
- ***Introduction of a neutral third party or enhancement of existing coordination centres:*** The different interests held by stakeholders – especially reporters and vendors – as well as the growing complexity of the landscape, both in terms of stakeholders and products, advocates the introduction of a neutral third party to coordinate vulnerability disclosure. An alternative is to enhance existing coordination centres, to ensure that power discrepancies as well as potential conflicts of interest will not compromise the overarching goal of improved information security.
- ***European policy makers and Member States should improve the legal landscape:*** The legal implications present many challenges for different stakeholders involved in the vulnerability disclosure process. The legal landscape must facilitate the improvement of information security and existing legislation must be adapted according to the latest developments.
- ***Vendors should facilitate trust building, transparency and openness*:** From a vendor's perspective, the stigma associated with acknowledging that one of its products contains a vulnerability could potentially lead to an unwillingness to recognise the existence of vulnerabilities. Society should therefore move towards a state where the existence of vulnerabilities is acknowledged and accepted, to facilitate more openness as a precursor to improved information security.
- ***ENISA should facilitate and advise on ways to improve the vulnerability disclosure landscape:*** ENISA could play a facilitating and advisory role in the area of vulnerability disclosure through information dissemination, providing recommendations, striving for harmonisation, collaborating with the security researcher community, and demonstrating leadership. From a policy perspective, ENISA could advise the European Commission about the necessity for transparency from vendors and the potentially negative impact of copyright law in the EU.

With our increasing dependence on ICT in everyday life, the exploitation of vulnerabilities discovered in information systems and networks will continue to pose grave security risks, with potentially damaging economic and societal impacts. Therefore it becomes all the more important that the various stakeholders involved in this complex environment attempt to address the various challenges that are encountered in the vulnerability disclosure landscape together. From the perspective of the security and trust of the end-users of systems, there is consensus that vulnerabilities must be disclosed in a way that minimises damage. Although movement towards more and better coordinated vulnerability disclosure has been happening to some extent for several years now, the landscape is still fragmented in many ways, and there are pertinent questions that remain unanswered. We explore and address some of these issues in this report. The analysis we present will be useful for all stakeholders involved in the disclosure of vulnerabilities, including vendors, researchers, policy makers, Computer Security Incident Response Teams (CSIRT) as well as the general public.

1. Introduction

1.1 Background of the study

Vulnerabilities are ‘flaws’ or ‘mistakes’ in computer-based systems that may be exploited to compromise the network and information security of the affected systems. They provide a point-of-entry or gateway for malicious activities and as such pose several, potentially severe security risks. Remedying vulnerabilities is therefore crucial and the disclosure process is a significant element that cannot be underestimated. Over the last few years, there has been a rise in the number of vulnerabilities that have been disclosed to the general public. In a number of cases, the vulnerabilities that were eventually reported were critical in terms of their potential or actual impact. In 2014, for example, several ‘high profile’ vulnerabilities were reported to the public, often via the media. These included, among others, Heartbleed, POODLE, Shellshock and Sandworm.[1] These disclosures carried several consequences and also re-opened the discussion on vulnerability disclosure procedures in general. Cindee Tran, for example, describes how in the case of the Shellshock vulnerability ‘*within hours of the release of this bug to the general public, attackers reportedly exploited this vulnerability to create botnets on compromised computers to perform DDoS (distributed denial-of-service) [attacks] and vulnerability scanning*.’[2] This led the security community to question whether the manner of disclosure was appropriate, especially since there may be a direct connection between the public disclosure of the vulnerability and the subsequent exploitation.

The vulnerability disclosure landscape is complex, with several stakeholders involved that include vendors, IT security providers, independent researchers, the media, malicious users and, ultimately, the general public. These stakeholders often have competing interests which results in a challenging landscape. Reflections on some of the challenges involved provide insight into the different perspectives within the community. Graham, for example, describes how ‘*Microsoft forced a self-serving vulnerability disclosure policy on the industry 10 years ago, but cries foul when Google does the same today*.’[3] One of the important points to note is that information security in general, and vulnerability disclosure in particular, ought not to become caught up in a ‘*battle of the giants*.’ As Böhme notes, ‘*as long as perfectly secure software is not available, the optimal distribution of vulnerability information is an important factor for the stability of a “network society”*.’[4]

Thus, the potentially sensitive nature of disclosing vulnerabilities poses a number of pressing questions which are associated with the existence of diverging and, at times, conflicting interests, as well as with legal restraints on the actions of stakeholders. Furthermore, digital and software-dependent technologies are becoming increasingly embedded in everyday life. Therefore it is vital for the economy and society at large that appropriate procedures are in place for disclosing vulnerabilities. Clearly, from the security perspective of end-users such as businesses and home users, vulnerability disclosure must occur, but who should do this, how it should be done, and when it is the right time to go public with vulnerabilities, are crucial questions that remain to be answered.

[1] Chapter 3 examines these four vulnerabilities in more detail.
[2] AppSec Consulting. 2015. ‘Zero-day Attacks in 2014.’ As of 7 October 2015: https://www.appsecconsulting.com/blog/zero-day-attacks-in-2014
[3] Errata Security. 2015. ‘A Call for Better Vulnerability Response.’ As of 29 May 2015: http://blog.erratasec.com/2015/01/a-call-for-better-vulnerability-response.html#.VWh4ms9VhBe
[4] Bohme, Rainer. n.d. ‘A Comparison of Market Approaches to Software Vulnerability Disclosure’ As of 29 May 2015: https://www.is.uni-muenster.de/security/publications/Boehme2006_CompVulnMarkets_ETRICS.pdf

The pursuit of these answers requires engagement with stakeholders to understand what challenges they face, what good practices they undertake, and what recommendations they have for improvement. This study is a first step to lead that process and provide further recommendations about what role ENISA as independent third party can play.

1.1.1 Objectives of the study

In the specific context of the complex vulnerability disclosure ecosystem, this study seeks to achieve the primary objectives listed in the following figure (Figure 2).

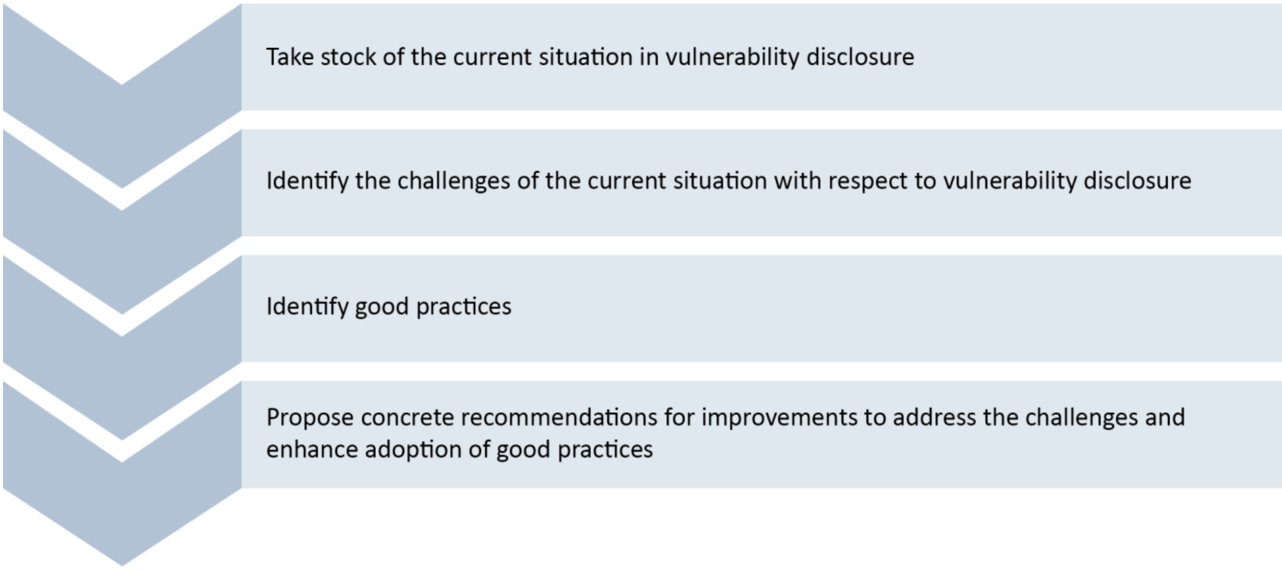


Figure 2: Primary objectives of the research

The analysis presented in this report will be useful for all stakeholders involved in the disclosure of vulnerabilities, including vendors, researchers, policy makers, Computer Security Incident Response Teams (CSIRTs) as well as the general public.

1.1.2 Methodology

In order to accomplish the objectives outlined above, the study relied on desk research and in-depth interviews with key experts in the field of vulnerability disclosure. A review of the available literature such as technical reports, company statements, media articles, and blogs, provided the basis for a general overview of the vulnerability disclosure ecosystem as outlined in chapter 2. In parallel, a set of case studies were prepared that focused on four widely distributed, critical vulnerabilities reported in 2014. For each vulnerability examined, the case study describes the whole lifecycle of the disclosure process from discovery to disclosure, as well as potential impact. The information for the case studies was gathered through keyword searches on Google, using the name of the vulnerability (e.g. Shellshock) and combinations of search terms that included disclosure, vulnerability, response and timeline. Sources used for the case studies mainly included media reports, advisories, and blog posts issued by journalists and vendors.

Both the literature review and the analysis of the case studies built the basis for the core task of the project, specifically the identification of challenges, good practices and the development of recommendations. In addition the research team conducted a total of 16 interviews during the course of the project and the team also received written responses from four additional participants who could not be interviewed in person. To obtain a range of opinions and insights, the interviewees were selected among experts across the vulnerability disclosure landscape and included vendors, information security companies, academic experts, co-ordinators (e.g. representatives from national computer security incident response teams) and civil society representatives.

The interviews were used to validate some of the findings from the literature review, and the information obtained in the interviews were also used to support the four case studies. The interviews which lasted approximately 45-60 minutes followed a semi-structured format (meaning that not all questions were covered at each interview). The complete list of interviewees is provided in Annex B and the indicative interview protocol is provided in Annex C.

To ensure that the interviewees were able to speak freely about their experience and knowledge, the study team does not attribute specific contributions to the interviewees in order to ensure confidentiality and to safeguard the sensitive nature of the topic discussed.

1.1.3 Outline of the report

In chapter 2 we present an overview of the vulnerability landscape, focusing on the definitions of vulnerabilities, statistics related to reported vulnerabilities, and key stakeholders involved in the disclosure process. Chapter 2 also looks at the different ways in which vulnerabilities can be disclosed. Four case studies of vulnerabilities reported in 2014 are presented in chapter 3. For each of the cases (Heartbleed, POODLE, Sandworm and Shellshock) the analysis covers the whole lifecycle of the disclosure process. Chapter 4 investigates the various challenges associated with the disclosure of vulnerabilities. A series of good practices are presented in chapter 5. The various sources of evidence are analysed in chapter 6, and a list of recommendations for improvement are presented. Finally, chapter 7 presents some concluding remarks and highlights some unanswered questions and strategic issues raised by the findings.

2. Overview of vulnerability landscape

2.1 Introduction

This chapter examines the overall landscape of vulnerability disclosure, by defining the term 'vulnerability' and by providing a brief background on the number of reported vulnerabilities (based on historical data contained in the National Vulnerability Database[5]). This is followed by an analysis of the different stakeholders involved in vulnerability disclosure. The chapter ends by examining the different ways in which vulnerabilities can be disclosed.

2.2 What is a vulnerability?

Any discussion of vulnerability disclosure must begin with a description of what constitutes a security vulnerability. Table 1 lists the definitions of 'vulnerability', compiled from a range of independent sources. Some definitions are more specific than others; for example, the Microsoft and Symantec definitions draw a direct connection between a vulnerability and the fundamental principles of information security, i.e. confidentiality, integrity and availability. In general, the various definitions refer to the following three key characteristics: (i) the existence of a 'flaw' or a 'weakness' (i.e. a 'point of entry') in a system; (ii) the ability of a potential threat (e.g. hackers) to have access the flaw; and (iii) the resulting ability to exploit it that may result in an information security compromise. These main elements of a vulnerability, which could potentially compromise the security of both software and hardware systems, have also been illustrated in Figure 3. Examples of the repercussions of vulnerabilities include the unauthorised remote execution of commands, denial of service attacks[6] and in extreme cases the complete shutdown of systems. It is worth reiterating that there are different degrees of vulnerability severity and associated risks which help to, for example, prioritise the way in which a detected vulnerability is 'fixed.' This is discussed in more detail in the next section.

Source	Definition
Committee on National Security Systems (CNSS) [Instruction No. 4009]	Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source.[7]
Common Vulnerabilities and Exposures (CVE)	An information security 'vulnerability' is a mistake in software that can be directly used by a hacker to gain access to a system or network. CVE considers a mistake a vulnerability if it allows an attacker to use it to violate a reasonable security policy for that system (this excludes entirely 'open' security policies in which all users are trusted, or where there is no consideration of risk to the system).[8]

[5] National Institute of Standards and Technology. n.d.– a. 'National Vulnerability Database.' As of 30 September 2015: https://nvd.nist.gov/
[6] CVE. 2013. 'Terminology.' As of 9 August 2015: https://cve.mitre.org/about/terminology.html
[7] Committee on National Security Systems (CNSS). 2010. 'National Information Assurance (IA) Glossary.' As of 9 August 2015: www.ncsc.gov/publications/policy/docs/CNSSI_4009.pdf
[8] CVE. 2013. https://cve.mitre.org/about/terminology.html

ENISA	The existence of a weakness, design, or implementation error that can lead to an unexpected, undesirable event compromising the security of the computer system, network, application, or protocol involved.[9]
Internet Engineering Task Force (IETF) Request for Comments (RFC) 2828	A flaw or weakness in a system's design, implementation, or operation and management that could be exploited to violate the system's security policy.[10]
Microsoft Security Response Centre (MSRC)	A security vulnerability is a weakness in a product that could allow an attacker to compromise the integrity, availability, or confidentiality of that product.[11]
National Cyber Security Centre (NCSC), the Netherlands	A vulnerability presents a hostile actor with the opportunity to inflict damage at a point where the protection against such damage is weak.[12]
National Institute of Standards and Technology (NIST) Special Publication 800-37	Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source.[13] [Same as CNSS definition.]
Symantec	A vulnerability is a weakness that allows an attacker to compromise the availability, confidentiality, or integrity of a computer system.[14]
Techopedia	Vulnerability is a cyber-security term that refers to a flaw in a system that can leave it open to attack. A vulnerability may also refer to any type of weakness in a computer system itself, in a set of procedures, or in anything that leaves information security exposed to a threat.[15]

Table 1: Contending definitions of a vulnerability

[9] ENISA. n.d. 'Glossary.' As of 9 August 2015: http://www.enisa.europa.eu/activities/risk-management/current-risk/risk-management-inventory/glossary#G52
[10] Network Working Group. 2000. 'Internet Security Glossary.' As of 9 August 2015: https://tools.ietf.org/html/rfc2828
[11] Microsoft Developer Network. n.d. 'Definition of a Security Vulnerability.' As of 9 August 2015: https://msdn.microsoft.com/en-us/library/Cc751383.aspx
[12] National Cyber Security Centre (2013a). *Policy for Arriving at a Practice for ResponsibleDdisclosure*. As of 12 October: 2015: https://www.ncsc.nl/binaries/content/documents/ncsc-en/current-topics/news/responsible-disclosure-guideline/1/Responsible%2BDisclosure%2BENG.pdf
[13] National Institute of Standards and Technology. 2010. 'Guide for Applying the Risk Management Framework to Federal Information Systems: A Security Life Cycle Approach.' As of 9 August 2015: http://www.csrc.nist.gov/publications/nistpubs/800-37-rev1/sp800-37-rev1-final.pdf
[14] Symantec. n.d.-a. 'Vulnerability Trends.' As of 9 August 2015: https://securityresponse.symantec.com/en/uk/threatreport/topic.jsp?id=vulnerability_trends
[15] Techopedia. n.d. 'Vulnerability.' As of 9 August 2015: http://www.techopedia.com/definition/13484/vulnerability

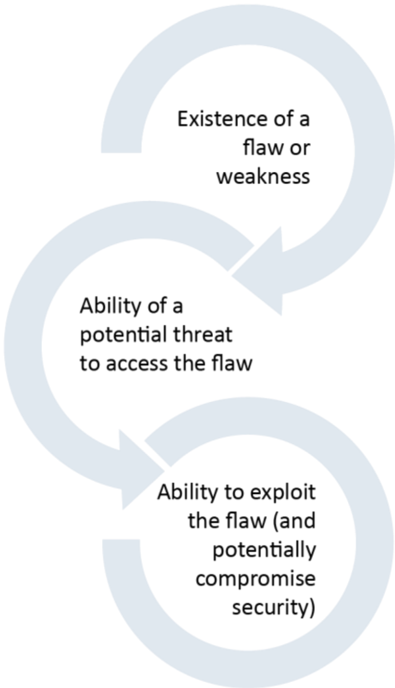


Figure 3: Underpinning characteristics of a security vulnerability

2.3 How many vulnerabilities are reported?

This section presents some high-level statistics about reported vulnerabilities based on data contained in the National Vulnerability Database (NVD). The NVD is a publicly accessible '*U.S. government repository of standards-based vulnerability management data [that enables] the automation of vulnerability management, security measurement, and compliance*.'[16] The NVD makes a comprehensive list of Common Vulnerabilities and Exposures (CVEs – see Table 1)[17] available to the public and attempts to quantify the risks posed by these vulnerabilities by 'scoring' CVEs based on a series of indicators (e.g. levels of impact and exploitability). In particular, the Common Vulnerability Scoring System (CVSS) facilitates a metrics-based technique for disseminating the different characteristics and impacts of vulnerabilities.[18] CVSS 'scores' are computed for almost all reported vulnerabilities. Based on the CVSS scores, the NVD also provides qualitative ratings of the severity of vulnerabilities using a 'Low-Medium-High' rating scale.[19] Specifically, vulnerabilities are categorised as low severity if they have a CVSS score between 0.0 and 3.9; medium severity vulnerabilities are those with CVSS scores between 4.0 and 6.9. Vulnerabilities are classified as high severity if they have a CVSS score between 7.0 and 10.0.

Figure 4 presents the historical trend (from 2001 onwards) in the annual number of security vulnerabilities reported, broken down by their respective CVSS 'Low-Medium-High' ratings. The relative proportions of low, medium and high severity vulnerabilities are presented in Figure 5. The graphs were generated using data obtained from the CVE statistics query page on the NVD website.[20] The number of vulnerabilities

---

[16] National Institute of Standards and Technology. n.d.- a. As of 29 May 2015: https://nvd.nist.gov/home.cfm
[17] CVE. 2015. 'About CVE.' As of 26 May 2015: https://cve.mitre.org/about/index.html
[18] National Institute of Standards and Technology. n.d.-b. 'CVSS v3 Information: NVD Common Vulnerability Scoring System Support v2.'As of 29 May 2015: https://nvd.nist.gov/CVSS.aspx
[19] National Institute of Standards and Technology. n.d.-b.
[20] The data were retrieved on 29 May 2015. . National Institute of Standards and Technology. n.d.-c. 'CVE and CCE Statistics Query Page.' As of 29 May 2015: https://web.nvd.nist.gov/view/vuln/statistics

corresponds to the number of 'software flaws' reported as per their CVE identifiers over the published date range 2001 to 2014. The yearly statistics are also listed in Annex A (both in terms of absolute numbers and proportions of the total number of vulnerabilities).

There was a general increase in the total number of vulnerabilities reported each year between 2001 and 2006, after which a steady downward trend was observed up to 2011. From 2011 onwards, however, the total number of vulnerabilities with assigned CVEs has shown an upward swing once again. Notably, the number of vulnerabilities reported in 2014 is substantially higher than the corresponding figure for 2013, peaking at just below 8,000 vulnerabilities, an 'all-time high.' The volume of vulnerabilities reported in 2014 represents a year-on-year increase of approximately 53% relative to 2013.

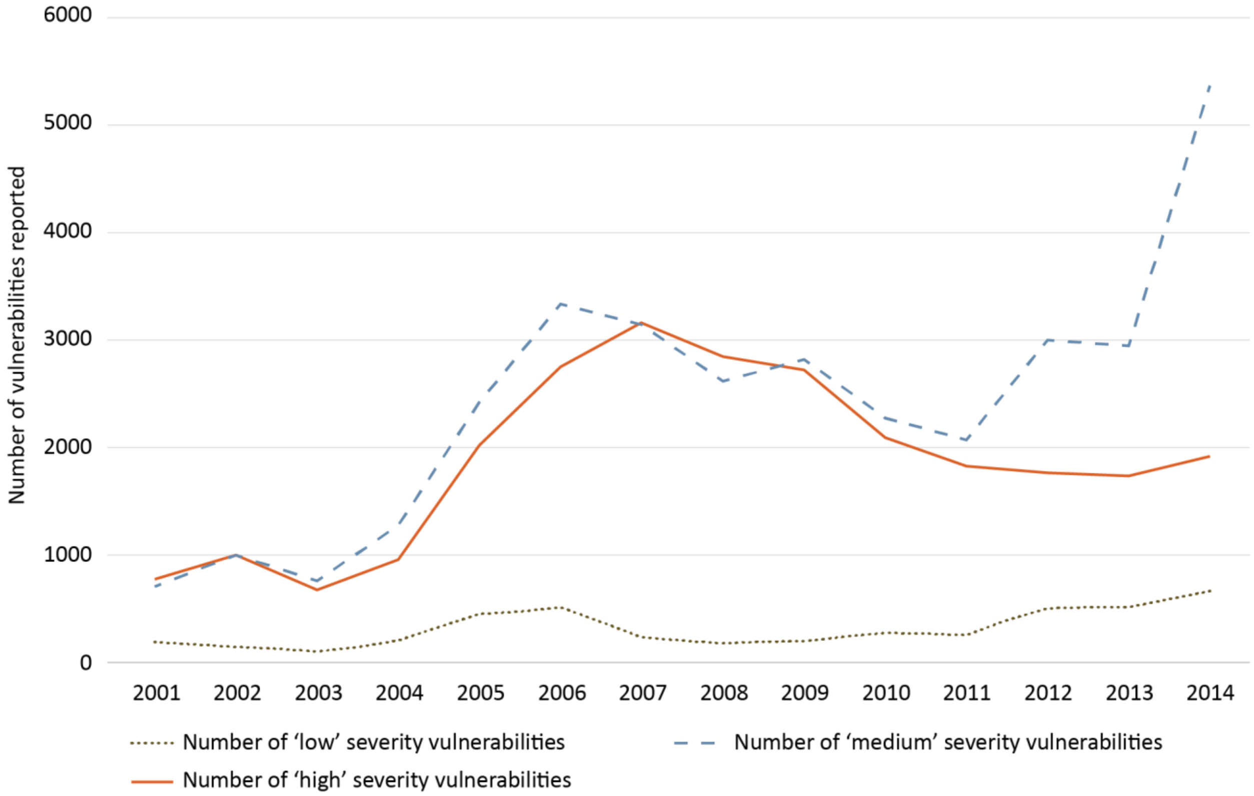


Figure 4: Yearly (2001-2014) distribution of the number of reported vulnerabilities broken down according to their severity 'rating'; the severity ratings are dependent on the computed NVD CVSS score[21]

The variation in the annual number of low and medium severity vulnerabilities displays a similar trend to the total number of vulnerabilities. The trend in the number of high severity vulnerabilities, however, exhibits a slightly different trend: there is a large increase from 2001 to 2007 (in which as many as 3,158 high severity vulnerabilities were reported, representing almost 50% of all vulnerabilities reported that year). After 2007, the number of high severity vulnerabilities reported shows a year-on-year decrease until 2013 in which 1,737 high severity vulnerabilities were reported. 2014 saw a slight reversal in this trend with over 1,900 high severity vulnerabilities published and a record high of around 5,356 medium severity vulnerabilities. The medium and high volumes together represented almost 92% of all reported vulnerabilities in 2014 (Figure 5). Interestingly, however, in 2014, the 1,920 high severity vulnerabilities constituted a proportion of slightly

[21] The graph was produced using CVE data retrieved from the National Vulnerability Database.

less than one quarter of all vulnerabilities reported that year. The 2014 figure for the percentage of high severity vulnerabilities is the lowest percentage across the entire period.[22]

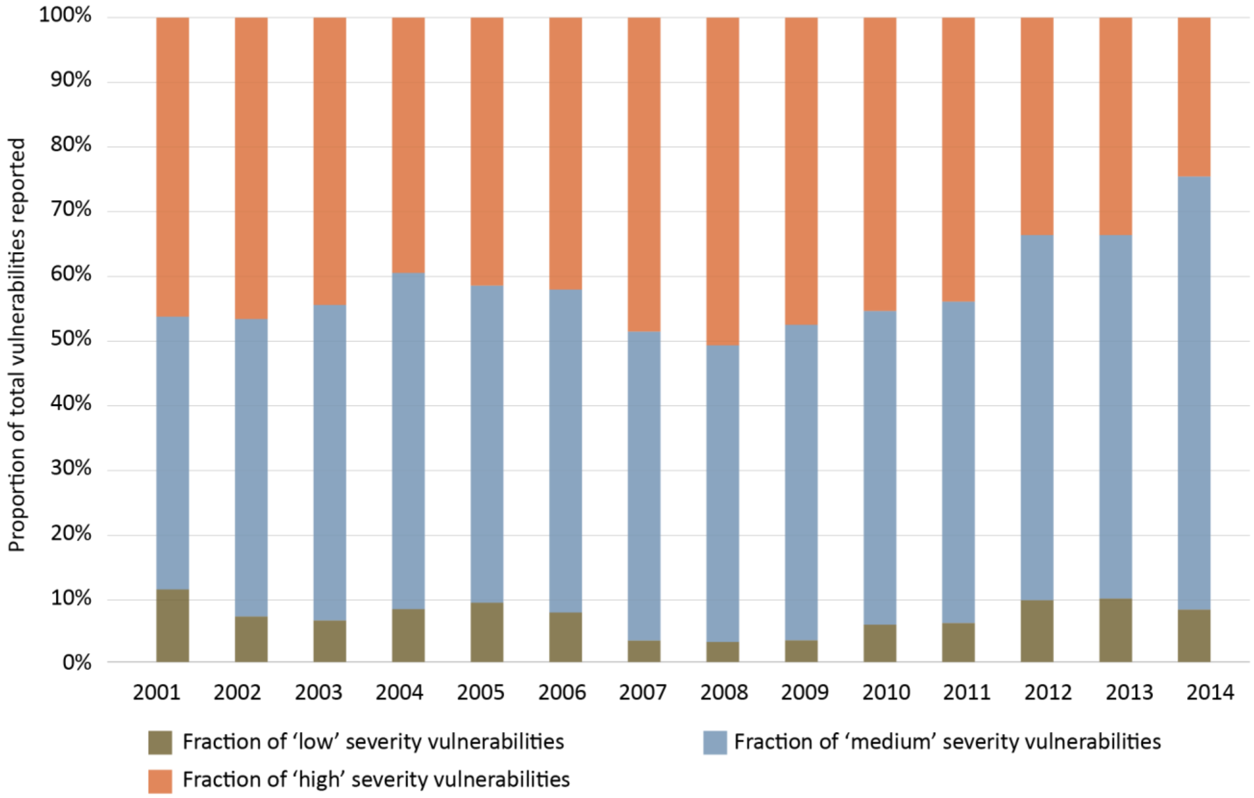


Figure 5: Yearly (2001-2014) distribution of the proportion of total vulnerabilities reported broken down according to their severity 'rating'; the severity ratings are dependent on the computed NVD CVSS score[23]

Finally, Figure 6 presents the yearly distribution of the number of vulnerabilities with CVSS scores exceeding 9.0.[24] Between 2001 and 2006 there was a steady increase in the number of these 'critical' vulnerabilities. In 2007 the number almost doubled to just under 1,000 vulnerabilities. Since 2007, the number of reported vulnerabilities has varied between approximately 800 and 1,100 each year, with only a very slight decrease in the overall number since 2012.

[22] The share of 'high' and 'medium' severity vulnerabilities varies across the whole period between approximately 89% and 97% of all reported vulnerabilities in the corresponding year.

[23] The graph was produced using CVE data retrieved from the National Vulnerability Database.

[24] Note: these data were retrieved from www.cvedetails.com, which relies on CVE vulnerability data taken from the NVD.

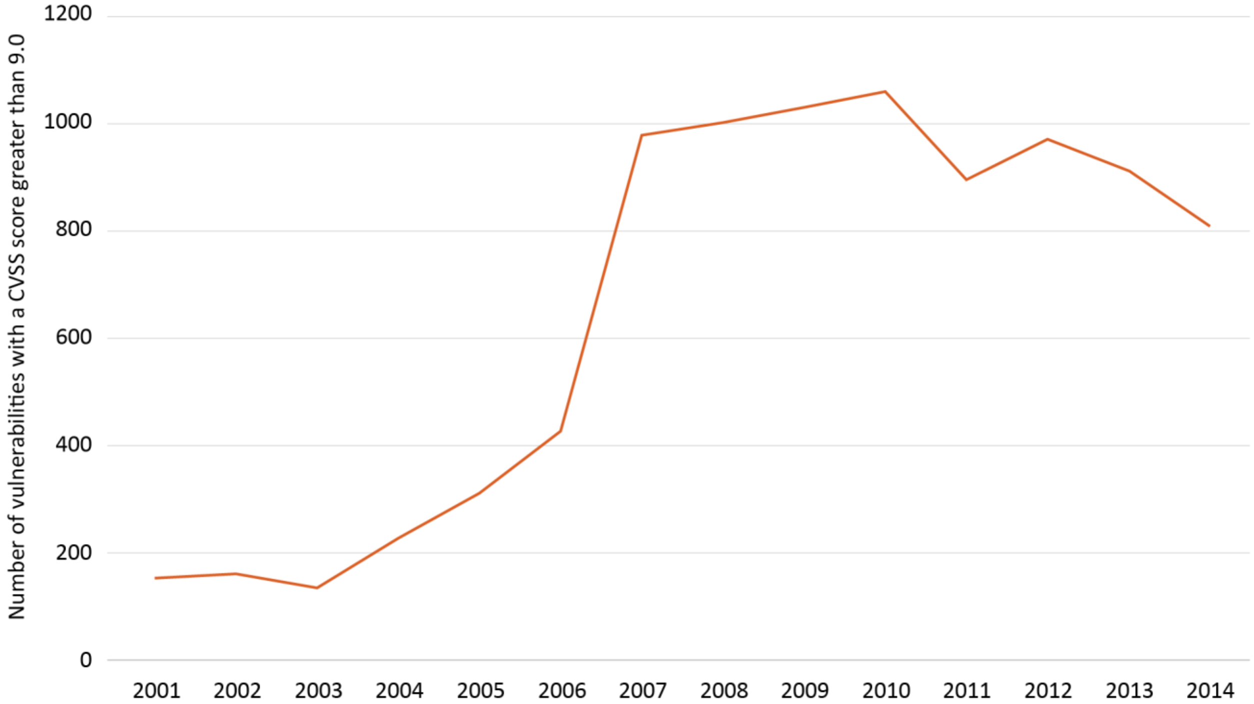


Figure 6: Yearly distribution (2001-2014) of the number of reported vulnerabilities with a CVSS score greater than 9.0[25]

It is important to note that these statistics provide only a broad indication of the historical trend in the disclosure of vulnerabilities based on assigned CVE identifiers. There are a number of limitations associated with simply counting the number of vulnerabilities that have been reported. The numbers do not indicate whether the vulnerabilities were eventually exploited by attackers, nor do they tell us whether the software flaws were ever patched by vendors.[26] Furthermore, for some vulnerabilities there is not enough information available to generate CVSS scores (e.g. a vendor discloses a vulnerability but does not provide sufficient data related to the bug). In these cases the NVD adopts a 'worst case approach' and allocates the maximum CVSS score of 10.[27] In addition, the NVD database is not comprehensive as not all vulnerabilities are assigned CVEs and are therefore not included in the data. Taking these limitations into account, the results need to be interpreted with caution.

Despite these caveats, the CVSS is a transparent attempt at quantifying the risks associated with vulnerabilities. By computing CVSS scores for the majority of reported vulnerabilities, the CVSS framework is able to provide a standard measure of the 'impact' of vulnerabilities for industries as well as governments. As noted on the NVD website, '*two common uses of CVSS are prioritization of vulnerability remediation activities and in calculating the severity of vulnerabilities discovered on one's systems.*'[28]

[25] The graph was generated using CVE data retrieved from CVE Details. [CVE Details. n.d. 'CVE and CCE Statistics Query Page.' As of 29 May 2015: http://www.cvedetails.com/cvss-score-charts.php?fromform=1&vendor_id=&product_id=&startdate=2001-01-01&enddate=2015-12-31&groupbyyear=1]
[26] Providing these data goes beyond the scope of this study.
[27] National Institute of Standards and Technology. n.d.-b.
[28] National Institute of Standards and Technology. n.d.-b.

2.4 Who's who in vulnerability disclosure?

The topic of vulnerability disclosure brings together several different stakeholders. This section aims to shed light on the 'who's who' in this field, but simultaneously emphasises how stakeholders may belong to various subgroups and that stakeholder groups are not necessarily mutually exclusive. In its National Infrastructure Advisory Council (NIAC) Vulnerability Disclosure Framework, the US Department of Homeland Security (DHS) identifies four major categories of stakeholder involved in the vulnerability disclosure process.[29] These include (i) **discoverers** (or finders); (ii) **coordinators**; (iii) **vendors**; and (iv) **users**. Figure 7 summarises the roles of these four primary participants.

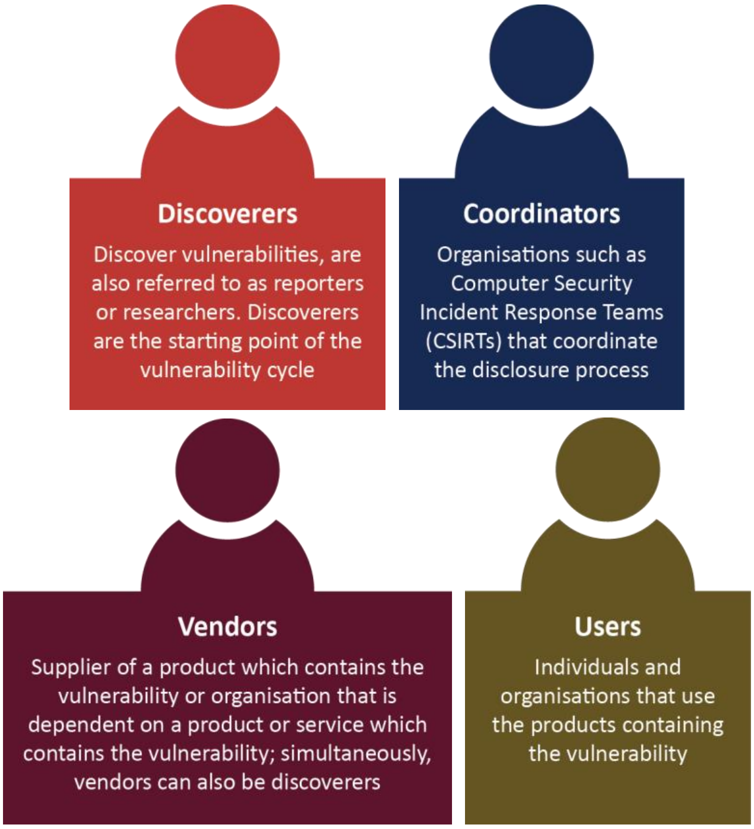


Figure 7: Primary stakeholder categories involved in the vulnerability disclosure process

Additional stakeholders that can be added to these broad categories include the **media**, **national governments**, more specifically law enforcement and national intelligence agencies, and profit-driven **criminals** (Figure 8).

[29] Chambers, John T. & John W. Thompson. 2004. 'Vulnerability Disclosure Framework.' As of 29 May 2015: http://www.dhs.gov/xlibrary/assets/vdwgreport.pdf

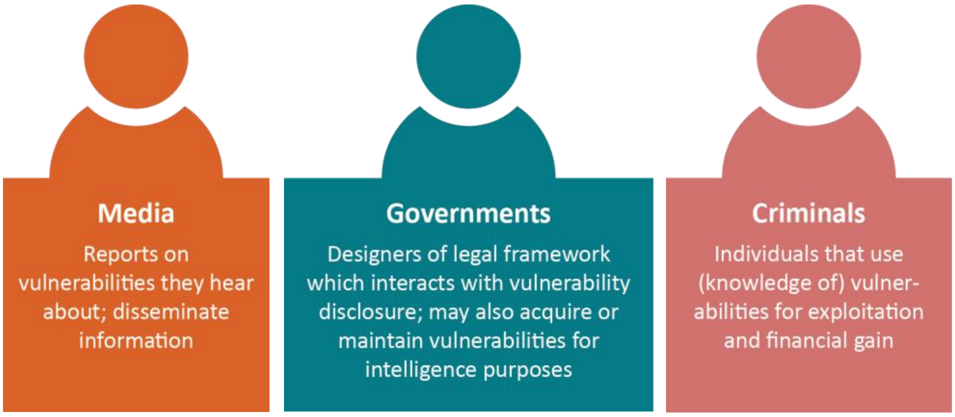


Figure 8: Additional stakeholder categories involved in the vulnerability disclosure process

There are stakeholders that are included in these groups but which may take on different roles or perspectives. Examples include vendors or information security companies which engage in vulnerability reward programmes, or academics who may be classified as discoverers, depending on the role they play. As such, this list is not meant to be exhaustive, but aims at highlighting the main roles. For the purposes of this report, a stakeholder is any individual, or group of individuals, which has a role in or influences the vulnerability disclosure landscape.

2.4.1 Vulnerability disclosure lifecycle and associated roles

The vulnerability disclosure lifecycle commences with a discovery of a flaw. The individual or organisation that discovers or finds the vulnerability is referenced as discoverer, often also as researcher (yet researcher is a more specific term and may not include all types of discoverers). On a sidenote: the discoverer may also be different from the reporter[30], mentioned later in this chapter. From the moment of discovery, the discoverer has a number of options on how to handle the vulnerability. First, the discoverer can report the vulnerability to the vendor. Christey & Wysopal (2002) define a vendor as '*an individual or organization who provides, develops, or maintains software, hardware, or services, possibly for free*.'[31] Vendors often maintain research staff who discover vulnerabilities and, as such, can potentially fulfil the role of discoverer in one scenario and the role of vendor in another. This shows the diversity within each stakeholder category, since a discoverer within a vendor may have a different interest or perspective compared to an independent researcher.

A discoverer can also report the vulnerability to a coordinator. This can be the discoverer's first action, or the discoverer may wish to reach out to a coordinator after the vendor ignores the report. A coordinator can be thought of as an intermediate entity that works with the discoverer/reporter and the vendor to 'fix' the vulnerability. As Christey & Wysopal (2002) note, '*Coordinators are often well-known third parties. Coordinators may have resources, credibility, or working relationships that exceed those of the reporter or vendors. Coordinators may serve as proxies for reporters, help to verify the reporter's claims, resolve conflicts, and work with all parties to resolve the vulnerability in a satisfactory manner*.'[32] Examples of coordinators used in the past are for example some Computer Security Incident Response Teams (CSIRTs).[33]

---

[30] As Christey & Wysopal (2002) note, 'A Reporter is the individual or organization that informs (or attempts to inform) the Vendor of the vulnerability. Note that the Reporter may not have been the initial discoverer of the problem.' [Christey, Steve & Chris Wysopal. 2002. 'Responsible Vulnerability Disclosure.' As of 12 August 2015: https://tools.ietf.org/html/draft-christey-wysopal-vuln-disclosure-00#page-3]
[31] Christey, Steve & Chris Wysopal. 2002.
[32] Christey, Steve & Chris Wysopal. 2002.
[33] The term CSIRT and Computer Emergency Response Teams (CERTs) are generally used interchangeably.

Coordinators can also represent a diverse category: on the one hand, coordinators can be, as mentioned, CSIRTs that are able to use their networks and 'central' position to capture the attention of the vendors and ensure vulnerabilities that are critical and reported receive an appropriate response. Also other types of coordinator exist, which have a commercial interest (examples will be discussed in the section on bug bounty programmes).

The discoverer can also decide to publicise the vulnerability more broadly, without informing the vendor or an intermediary, and directly publish the information to users of a product. (The Department of Homeland Security includes users as a stakeholder in its categorisation as the manner in which the vulnerability is disclosed and the subsequent potential and likelihood for exploitation ultimately affect their level of security.[34])

From the moment a potential security vulnerability is discovered, there are a number of basic steps involved in the process of disclosing the vulnerability to the public. The primary steps, illustrated and summarised in Figure 9, include the following: (i) **discovery**; (ii) **notification**; (iii) **investigation**; (iv) **resolution**; and (v) **release**.[35]

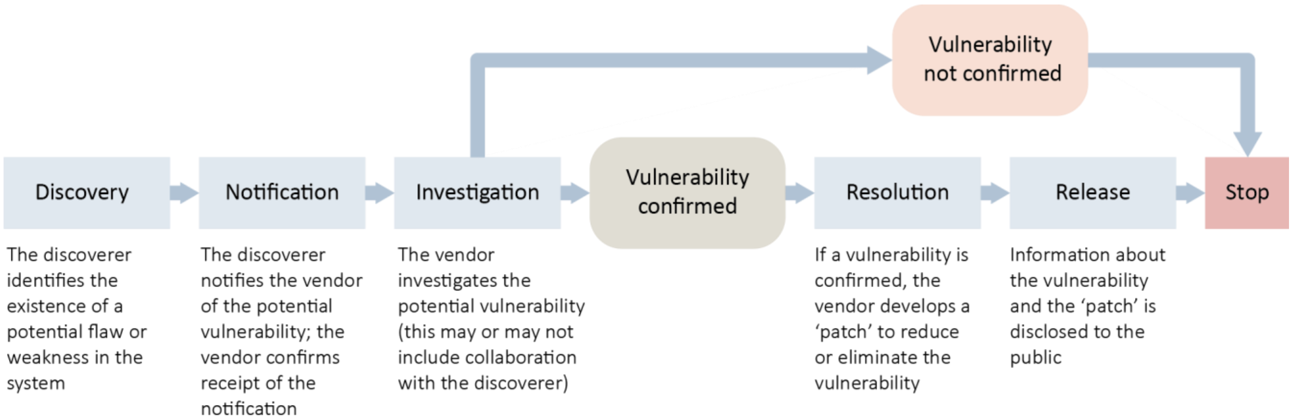


Figure 9: Key steps involved in the disclosure of security vulnerabilities[36]

2.4.1.1 Role of the media

The media fulfils a crucial role in the dissemination of information with regard to vulnerability disclosures. Hill (2014), for example, describes how in the case of the GoToFail vulnerability and the '*Apple silence*', journalists and security researchers had to inform the public and respond to its questions regarding the vulnerability.[37] The media, as a result, can be perceived to fulfil a number of roles, which may not be captured in a more traditional model of vulnerability disclosure processes as described above.

The treatment of vulnerabilities by the media has for sure influenced the vulnerability disclosure landscape, sometimes in a good way, sometimes not (according to the experts interviewed during this study).

Interviewees indicated how – before answering the questions – distinctions between media outlets as well as vulnerability reporters must be made since the level of technical knowledge influences the quality of

[34] Chambers, John T. & John W. Thompson. 2004.
[35] Organization for Internet Safety. 2004. 'Guidelines for Security Vulnerability Reporting and Response.' As of 30 September 2015: https://www.symantec.com/security/OIS_Guidelines%20for%20responsible%20disclosure.pdf
[36] Adapted from: https://www.symantec.com/security/OIS_Guidelines%20for%20responsible%20disclosure.pdf
[37] Hill, Kashmir. 2014. 'Apple's Deafening Silence on 'GoToFail' Security Flaw.' Forbes, February 24. As of 30 September 2015: http://www.forbes.com/sites/kashmirhill/2014/02/24/apples-deafening-silence-on-gotofail-security-flaw/

reporting. Information security media journalists and outlets, for example, generally maintain a high level of knowledge and expertise, and by this are in a better position to report accurately on an issue. A general perception, however, is that media reporting is prone to 'hyping' vulnerabilities, especially as a result of a trend to label vulnerabilities and market them with catchy names and logos. The hype may overshadow the technical characteristics of the vulnerability and as such worry the public and other stakeholders unnecessarily. Figure 10 presents a snapshot of some of the headlines that appeared in online media relating to the case studies included in this report (see Chapter 3).

Figure 10: Example headlines in the global online media relating to the vulnerability case studies included in this report

The positive influence of the media is that the attention devoted to information security in general, and vulnerability disclosure in particular, has made it a relatively mainstream topic. Interviewees recognised this as well, describing how the media highlights and pushes the issue of cybersecurity, and improves security awareness among the public. The media is in a position to reach out to the general public, which can have two other benefits. Firstly the ability to exert pressure on vendors to take the topic of vulnerability disclosure seriously. Full disclosure of a vulnerability only has an impact as long as the media is willing to report on the issue. The second aspect is the ability of the media to inform the public when a vulnerability requires action on the part of the end-user such as installing a patch.

2.5 Different forms of vulnerability disclosure

How, when, and by whom a vulnerability is disclosed are core topics of debate within the information security community. Disclosure of vulnerabilities can occur roughly in three different ways. Cencini et al.

(2005) identify three different types of vulnerability disclosure.[38] The first is *non-disclosure* which is the easiest to explain but the most difficult to quantify. Non-disclosure means that the discoverer keeps the vulnerability secret and does not report it, neither to the vendor nor to the public (Figure 11). Non-disclosure may, however, include the sale of the vulnerability to a third party.

Figure 11: Non-disclosure of vulnerabilities

The second form is *full disclosure*, which is the opposite of non-disclosure since with full disclosure the discoverer discloses the vulnerability to the public at large (Figure 12). Full disclosure may not give the vendor sufficient time and warning to resolve the vulnerability and makes vulnerability information openly available to malicious potential attackers. Full disclosure does not discriminate among its recipients or audience.

Figure 12: Full disclosure of vulnerabilities

The third form is *responsible disclosure*. This is an ambiguous term, especially since it has a normative connotation of being ‘responsible’ and interpretation varies from stakeholder to stakeholder. As a result the more recent preference is to use the term *coordinated vulnerability disclosure*, as it’s perceived as being more neutral. In both cases, the discoverer reports the vulnerability to the vendor with the intention of assisting the vendor in resolving the vulnerability (Figure 13). When a resolution is available, the vendor publishes the vulnerability alongside a patch for users. The knowledge about the vulnerability does not become public until a solution is available.

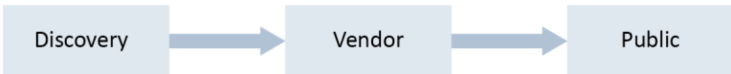


Figure 13: Responsible or coordinated disclosure of vulnerabilities

There is also *limited disclosure*, which can also be classified as a variant of responsible and coordinated vulnerability disclosure. Limited disclosure occurs when only specific parties are informed about a vulnerability. Coordinated vulnerability disclosure can also include the presence of a third party to lead the coordination process (Figure 14).

[38] Cencini, Andrew, Kevin Yu, and Tony Chan. 2005. ‘Software Vulnerabilities: Full-, Responsible-, and Non-Disclosure.’ As of 9 October 2015:
http://courses.cs.washington.edu/courses/csep590/05au/whitepaper_turnin/software_vulnerabilities_by_cencini_yu_chan.pdf

Figure 14: Responsible disclosure of vulnerabilities involving a coordinator

2.5.1 Different opinions about disclosure types and the challenges associated

A variety of arguments exist for and against the adoption of the different existing vulnerability disclosure forms. Full disclosure is perceived as objectionable since the software vendor(s) may not be aware of the vulnerability or may not have enough time to develop a solution for it before the disclosure becomes public. Once the vulnerability has been disclosed to the public, and a solution to patch the vulnerability is not available (yet), malicious actors are in a position to exploit it.[39] This claim against full disclosure, however, has been criticised by other experts. According to Schneier, full disclosure is a '*good idea*.' Schneier writes: '*Public scrutiny is the only reliable way to improve security, while secrecy only makes us less secure*.'[40] The fear that malicious actors may exploit the information provided through full disclosure assumes that they are not already aware of the vulnerability. The other assumption is that software vendors will indeed spend time and financial resources to fix the vulnerability. For Schneier, both of these assumptions are false. Schneier delves into the economics of the matter to explain how there has to be an incentive for software vendors to fix the vulnerability, which is not necessarily always present. Cavusoglu et al., however, write '*[a]lthough immediate public disclosure might provide necessary motivations to vendors that might otherwise ignore the vulnerability otherwise [sic], it also punishes other vendors that would make an honest effort to deliver the patch promptly by not providing them adequate time to address the vulnerability*.'[41]

The greatest challenge in the area of vulnerability disclosure is the fundamental difference in opinions among stakeholders and stakeholder groups. This is precisely why the vulnerability disclosure debate, despite going on for ages, remains alive. There is an inherent difference of opinion between discoverers, who prefer to advocate a full disclosure approach, and vendors, who prefer to advocate a 'responsible' or coordinated vulnerability disclosure approach. As an interviewee stated, '*The reality is that they are actually both right in their own way. They are both right in terms of their own worst case scenario*.' The challenge then is to deviate from an extreme point of view and to build in an inherent flexibility which allows one's perspective to change according to the specifics of a vulnerability. Not every vulnerability, after all, will fall into the category of the worst case scenario.

Moreover, attention needs to be devoted to different classes of attack.[42] There are opportunistic attacks which generally rely more heavily on publicly available information about vulnerabilities and exploits. Perpetrators of these types of attacks are willing to carry out a high number of attacks each with a low probability of success but a high probability of detection. Attacks of this nature are measurable and their impact can be quantified to a certain extent. This is in contrast to targeted attacks, where perpetrators prefer custom-developed solutions, have a base level of operational security, and have access to a range of software exploits. Moreover, a specific feature of targeted attacks is that the perpetrator has a strategic target in mind. These types of attacks are difficult or potentially even impossible to measure in a reliable manner. How individuals think about vulnerability disclosure policy, however, is influenced by their ability to model and anticipate targeted attacks and associated behaviour. If a user, whether an individual or an organisation, does not have the ability to model targeted attack behaviour – perhaps due to a lack of high-quality data – then there is a bias towards a longer public release timeline for the vulnerabilities. As one interviewee notes, 'These decisions are designed to reduce harm from opportunistic attacks and assume that no other attackers have access to the vulnerability information.' Based on observations from the private

[39] See for example Cavusoglu, Hasan, Huseyin Cavusoglu & Srinivasan Raghunathan. n.d. 'Emerging Issues in Responsible Vulnerability Disclosure.' As of 7 October 2015: http://www.infosecon.net/workshop/pdf/65.pdf)
[40] Schneier, Bruce. 2007. 'Schneier: Full Disclosure of Security Vulnerabilities a "Damned Good Idea".' As of 7 October 2015: https://www.schneier.com/essays/archives/2007/01/schneier_full_disclo.html
[41] Cavusoglu, Hasan, Huseyin Cavusoglu & Srinivasan Raghunathan. n.d.
[42] Written input from respondent.

exploit market, there appears to be a growing number of vulnerability discoveries that are already known by advanced attackers. This is a phenomenon known as 'bug collision.' When a vulnerability is already known by advanced attackers, the level of urgency increases and timelines for public release are shortened.

Arguably many efforts and initiatives that are being introduced aim to facilitate the development of a common ground to overcome the existing fundamental difference of opinion. Yet, based on the discussion on types of attacks, it becomes evident that disclosure ought to be tailored to the vulnerability and that polarised positions should be replaced by a more flexible perspective.

2.5.2 Role of CSIRTs

Direct communication between software vendor and vulnerability reporters may be hampered by distinct interests both parties hold, as well as the fundamental differences of opinion described above. This is evident both with respect to independent discoverers as well as with regard to security researchers who may be employed by a vendor and who report the vulnerability to another vendor. This tension leads to the introduction of CSIRTs as a potentially important stakeholder that can play a role in coordinating the process and keeping the public interest at heart. Within the information security community, CERT Coordination Centre (CERT-CC)[43] has held a vulnerability disclosure process for many years. The tradition of CSIRTs acting as a coordinator appears to be a more recent development in the European Union (EU), although CERT-FI[44] was broadly recognised by interviewees as having significant experience in this role. CIRCL, the CSIRT for the private sector, communes and non-governmental entities in Luxembourg, also maintains a vulnerability disclosure policy.[45]

Through knowledge of the vulnerability, CSIRTs are also in a position to (confidentially) notify their constituents, such as critical infrastructure organisations, to make them aware of the vulnerability and potentially take steps ahead of the release of a patch. In this way, the vulnerability is disclosed, but to a select group to ensure malicious actors do not obtain the damaging information.

During the interviews the project team specifically asked the interviewees what role they believed CSIRTs can play in this area. This section must be prefaced by noting that certain CSIRTs already play an active role in coordinated vulnerability disclosure. These include – as noted above – CERT-CC based in the United States, CERT-FI based in Finland and JP-CERT based in Japan. Moreover, the National Cyber Security Centre (NCSC) in the Netherlands also maintains a role in vulnerability disclosure as it launched a policy in 2013.[46]

Opinions of the role of CSIRTs differ, depending on the experience of the interviewees. Some interviewees were sceptical about CSIRTs, mainly due to the CSIRTs' direct 'government connection or association' (see 4.2.5). Interviewees expressed concerns about the potential implications of sharing vulnerability information with an organisation which is 'part of the government' and may not be 'independent.' Such scepticism varied, based on which nation or Member State respondents were speaking of and what their experiences were with respect to CSIRTs. Other respondents, for example, very clearly indicated that such claims were unjustified based on their experiences. As indicated by one CSIRT, in its experience the reporter needs to

[43] The CERT Coordination Center (CERT/CC) is the Coordination Centre of the Computer Emergency Response Team (CERT) for the Software Engineering Institute (SEI), a non-profit US federally funded research and development centre.
[44] CERT-FI is the Finnish national computer security incident response team whose task is to promote security in the information society. CERT-FI is a part of the Finnish Communications Regulatory Authority.
[45] Computer Incidence Response Center Luxemburg (CIRCL). 'Responsible Vulnerability Disclosure.' As of 7 October 2015: https://www.circl.lu/pub/responsible-vulnerability-disclosure/
[46] National Cyber Security Centre. 2013b. 'Responsible Disclosure Guideline.' As of 7 October 2015: https://www.ncsc.nl/english/current-topics/news/responsible-disclosure-guideline.html

feel a certain level of confidence with regard to the CSIRT in its vulnerability coordinator role. Publicly publishing a disclosure policy helps to gain a certain level of trust from the reporter.

The main role which CSIRTs can play with respect to vulnerabilities is as a 'coordinator' of the process. This could, however, become complicated as expressed by the scepticism of some of the interviewees. As coordinators of the process, CSIRTs may become involved in certain parts of the vulnerability disclosure processes, but not all. The CSIRTs that the project team spoke to indicated how they generally encouraged discoverers to first try to contact the vendor themselves before approaching the CSIRT. Discoverers could reach out directly to CSIRTs in case they wanted to remain anonymous, did not have the contact information for the vendor, or if the vulnerability involved multiple vendors or organisations. Discoverers may also contact CSIRTs if they cannot come to an agreement with the vendor or the vendor ignores their reports.

As one interviewee described it, 'The dynamic from the software company will change if a third party is involved. So if it's just a relationship between the vulnerability researcher and the software company, there is a higher chance that the software company will be confrontational, and a lower chance that they will resolve the issue in a timely manner.'

2.5.3 Bug bounty programmes reward reporters

The relationship between discoverers and vendors has evolved over time, but remains complex since they have different objectives and interests, and as such also often distinct opinions on the severity of the vulnerability and the necessity to inform the public, and how fast this should be done, if at all. One of the main developments in bringing vendors and discoverers closer together has been the introduction of bug bounty programmes, Vulnerability Purchase Programmes (VPPs) and Vulnerability Rewards Programmes, which have all become commonplace. A bug bounty programme rewards a reporter for reporting a discovered vulnerability. According to Friis-Jensen (2014), bug bounty programmes have existed since October 1995 when Netscape offered a cash reward to those who were able to find and subsequently report security vulnerabilities in Netscape Navigator 2.0 Beta.[47] The idea, however, was not adopted by other software vendors until several years later. IDefense, a company which VeriSign later acquired, introduced an initiative which offered researchers cash rewards for reporting software vulnerabilities.[48] IDefense functioned as an intermediary between the researchers reporting the vulnerabilities and the software vendors. TippingPoint in 2005 introduced the Zero-Day Initiative (ZDI), which was a competing programme to the initiative introduced by IDefense.[49] The main goals of the ZDI are:

- 'Extending TippingPoint's security research organisations (via DVLab's research team) by leveraging the methodologies, expertise, and time of others
- Encouraging the reporting of zero-day vulnerabilities in a responsible manner to the affected vendors by financially rewarding researchers
- Protect its customers through the TippingPoint Intrusion Prevention Systems (IPS) while the affected vendor is working on a patch'[50]

---

[47] Cobalt. 2014. 'The History of Bug Bounty Programs.' Cobalt, 11 April. As of 30 September 2015: https://cobalt.io/blog/the-history-of-bug-bounty-programs/
[48] See, for example, Itnews. 2007. 'Vista contest offers cash for exploits.' As of 25 October: http://www.itnews.com.au/news/vista-contest-offers-cash-for-exploits-71238
[49] Zero Day Initiative. n.d.-a. 'Why Did We Create the Zero Day Initiative?.' As of 7 October 2015: http://www.zerodayinitiative.com/about/
[50] Zero Day Initiative. n.d.-a.

On its website, the ZDI describes how the programme operates:[51] Researchers can approach the ZDI with exclusive information on unpatched, or zero-day, vulnerabilities. ZDI then conducts its own research and analysis to validate both the identity of the researcher as well as the vulnerability. ZDI proceeds to make a monetary offer to the researcher. If the researcher accepts, then the researcher receives a prompt payment. ZDI indicates how subsequently vendors are notified of the vulnerability, while Tippingpoint also develops an Intrusion Prevention System (IPS) filter. The ZDI maintains a specific policy for its vulnerability disclosure to vendors. The ZDI will try to contact the vendor at least three times via different means, i.e. email, phone and through an intermediary. However, '*[if] DVLabs exhausts all reasonable means in order to contact a vendor, then DVLabs may issue a public advisory disclosing its findings fifteen business days after the initial contact*.'[52]

Frei describes how both iDefense and the ZDI do not act on all vulnerabilities they receive, but rather appear to focus on highly critical vulnerabilities that target prevalent software products. As can be seen in Table 2, the number of public disclosures made by ZDI has seen an increase over the last three years.[53]

Table 2: Historical trend of the number of public disclosures made through the Zero-day Initiative[54]

Year	Number of disclosures
2012	203
2013	287
2014	427

Besides these 'intermediary initiatives', bug bounty programmes initiated by vendors have become more common since 2004. During that year the Mozilla Foundation introduced cash rewards to researchers who reported critical vulnerabilities in Firefox. The Mozilla bug bounty programme is still active and now covers most of its products.[55] Currently several other big names, such as Facebook, Google and Microsoft maintain bug bounty programmes. Facebook has its Whitehat programme, where researchers can report the bugs they discover. Depending on the severity of the detected vulnerability, Facebook will provide a cash reward of at least US$500, although there is no pre-determined maximum. The findings from its Whitehat programme are showcased on the Facebook Bug Bounty page. This page also reports on statistics of the programme during the previous year. For 2014, Facebook described how:

> *'Submissions increased by 16% to 17,011. We are happy to see that the program is continuing to produce high quality reports – 61 of last year's eligible bugs were categorized as high severity, 49% more than the previous year.*
>
> *We've paid out more than $3 million since we got started in 2011, and in 2014 we paid $1.3 million to 321 researchers across the globe. The average reward in 2014 was $1,788.*

---

[51] Zero Day Initiative. n.d.-a.
[52] Zero Day Initiative. n.d.-b. 'Disclosure Policy.' As of 7 October 2015: http://www.zerodayinitiative.com/advisories/disclosure_policy/
[53] Shannon_Sabens. 2015. 'Milestone today, good times ahead.' HP Security Research Blog, 12 May. As of 7 October 2015: http://h30499.www3.hp.com/t5/HP-Security-Research-Blog/Milestone-today-good-times-ahead/ba-p/6743824#.VWmwrU0tGUk
[54] Shannon_Sabens. 2015.
[55] Mozilla Security. n.d. 'Bug bounty program.' As of 12 October 2015: https://www.mozilla.org/en-US/security/bug-bounty/

65 countries received rewards this year, representing a 12% increase from 2013. We now have 123 countries reporting bugs.'[56]

Besides having individual programmes, certain organisations also contribute to more overarching initiatives such as the Internet Bug Bounty programme which is dedicated to the discovery of vulnerabilities in frameworks that are used in many applications.[57] Another initiative is HackerOne, a company which provides a platform designed to streamline vulnerability coordination and bug bounty programmes by enlisting hackers to improve vendor security.[58] It is important to emphasise that HackerOne itself is not a bug bounty programme, but rather works with vendors who have such programmes. HackerOne's services are free unless the vendor or another organisation pays the discoverer for the vulnerability, in which case HackerOne charges a commission. HackerOne currently has a wide-ranging clientele as well as a large number of researchers affiliated with the platform. According to Perlroth, HackerOne has about 1,500 users on its platform and collectively they have fixed around 9,000 vulnerabilities, earning around US$3 million in bounties.[59]

Overall, these bug bounty programmes have been welcomed by the information security community because they bridge the gap between discoverers and vendors. Through these programmes, vulnerability discovery has become a more structured and rewarding process for both parties, which is more conducive to effective cooperation. Even so, there is some reluctance with respect to the bug bounty programmes and paying for vulnerabilities in general (see 4.2.7).

[56] Facebook Bug Bounty. n.d. In *Facebook* Product/Service page. As of 7 October 2015: https://www.facebook.com/BugBounty
[57] Internet Bug Bounty. n.d. 'Internet Bug Bounty: Rewarding friendly hackers who contribute to a more secure internet.' As of 1 June 2015: https://hackerone.com/internet-bug-bounty
[58] HackerOne. n.d. As of 7 October 2015: https://hackerone.com/
[59] Perlroth, Nicole. 2015. 'HackerOne Connects Hackers With Companies, and Hopes for a Win-Win.' New York Times, 7 June. As of 7 October 2015: http://www.nytimes.com/2015/06/08/technology/hackerone-connects-hackers-with-companies-and-hopes-for-a-win-win.html?_r=0

More than Money: United Airlines Awards Miles[60,61]

In May 2015, United Airlines demonstrated an original variation of a bug bounty programme. Instead of money, United Airlines decided to reward reporters in 'award miles':

Severity	Examples	Maximum payout in award miles
High	Remote code execution	1,000,000
Medium	Authentication bypass Brute-force attacks Potential for personally identifiable information (PII) disclosure Timing attacks	250,000
Low	Cross-site scripting Cross-site request forgery Third-party security bugs that affect United	50,000

There are other curious features to the company's bug bounty programme, especially since it removed from a flight a researcher who tweeted about a vulnerability in the airline's Wi-Fi-system only weeks before the introduction of its programme. United Airlines specifies which vulnerabilities are eligible for submission and which are not. In light of events with respect to the tweeting researcher, the latter category is particularly worthy of mention since United Airlines considers the following vulnerabilities ineligible:

- 'Bugs that only affect legacy or unsupported browsers, plugins or operating systems
- Bugs on internal sites for United employees or agents (not customer-facing)
- Bugs on partner or third-party websites or apps such as:
  - cruises.united.com
  - hotels.united.com
  - hub.united.com
  - unitedmileageplus.com
  - vacations.united.com
- Bugs on onboard Wi-Fi, entertainment systems or avionics
- Insecure cookie settings for non-sensitive cookies
- Previously submitted bugs
- Self-cross-site scripting
- Vulnerabilities that apply only to you or your own account'[62]

2.5.4 Zero-day market

Besides bug bounty or vulnerability reward programmes, discoverers can also receive financial rewards via other channels. This can occur through the so-called zero-day market. Sources in the media have previously described aspects of the zero-day market about which they have been able to acquire information.

[60] United. n.d. 'United Airlines bug bounty program.' As of 7 October 2015: http://www.united.com/web/en-US/content/contact/bugbounty.aspx
[61] Zetter, Kim. 2015a. 'United Airlines Pays Man A Million Miles for Reporting Bug.' *Wired*, 14 July. As of 7 October 2015: http://www.wired.com/2015/07/united-airlines-pays-man-million-miles-reporting-bug/
[62] United. n.d.

Greenberg specifically focuses on a company named Vupen.[63] Bekrar, Vupen's chief executive features prominently in the article as his team 'dismantled' Google Chrome during a Google-sponsored contest. Vupen did not act as a participant in the contest, instead the company carried out its activities in the background. Google offered US$60,000 for the details of their approach, but Vupen refused. Greenberg quotes Bekrar who states: '*We wouldn't share this with Google for even $1 million. We don't want to give them any knowledge that can help them in fixing this exploit or other similar exploits. We want to keep this for our customers*.'[64] According to Greenberg, the clientele of Vupen consists of government agencies which acquire the vulnerabilities to infiltrate and disrupt the operations of criminals and intelligence targets. This situation has led to concerns among civil society groups.[65] Concerns exist not only because the zero-day vulnerabilities could end up in the wrong hands, but also because there may be a lack of oversight regarding the situations for which intelligence agencies and law enforcement officials use them.

Schneier describes how the market for zero-days has matured substantially over the last few years. He describes how in 2007 Charlie Miller described his attempts to sell zero-day exploits, and how in 2010 a survey indicated that the opportunity for a zero-day market was limited. Schneier simultaneously reflects on how the progression of a zero-day market functions as a game changer with dangerous consequences. As Schneier writes, '*I've long argued that the process of finding vulnerabilities in software systems increases overall security. This is because the economics of vulnerability hunting favoured disclosure*.'[66] The introduction of the zero-day market, however, changes these dynamics. He indicates how, since the grey market is more lucrative than the public vulnerabilities market, it will lead to more hackers choosing the former rather than the latter.

Zero-day vulnerabilities on the grey market are purchased on the assumption that the sale is exclusive and that the vendor will not be notified of the vulnerability.[67] Sometimes, according to Greenberg, the payment will be made in instalments as long as the vendor has not patched the vulnerability. A vulnerability after all only maintains its zero-day status as long as no solution is available and disseminated among the users of the product. Although then it could still have value, as indicated by Ablon et al., as a 'half-day.'[68]

The majority of interviewees who mentioned the issue of exploiting vulnerabilities for national intelligence purposes identified this as a challenge, especially since the zero-day market is able to offer significantly higher rewards to reporters than can more legitimate efforts such as bug bounty and vulnerability reward programmes.

[63] Greenberg, Andy. 2012. 'Meet the Hackers Who Sell Spies the Tools to Crack Your PC (And Get Paid Six-Figure Fees).' Forbes, 21 March. As of 7 October 2015: http://www.forbes.com/sites/andygreenberg/2012/03/21/meet-the-hackers-who-sell-spies-the-tools-to-crack-your-pc-and-get-paid-six-figure-fees/
[64] Greenberg, Andy. 2012.
[65] Hofmann, Marcia. 2012. '"Zero-day" exploit sales should be key point in cybersecurity debate.' Electronic Frontier Foundation (EFF), 29 March. As of 7 October 2015: https://www.eff.org/deeplinks/2012/03/zero-day-exploit-sales-should-be-key-point-cybersecurity-debate
[66] Schneier, Bruce. 2012. 'The Vulnerabilities Market and the Future of Security.' Forbes, 30 May. As of 7 October 2015 : http://www.forbes.com/sites/bruceschneier/2012/05/30/the-vulnerabilities-market-and-the-future-of-security/
[67] Greenberg, Andy. 2012.
[68] Ablon, Lillian, Libicki, Martin C. Golay Andrea A. 2014. 'Markets for Cybercrime Tools and Stolen Data: Hackers' Bazaar.' Santa Monica, Calif.: RAND Corporation. As of 7 October 2015: http://www.rand.org/content/dam/rand/pubs/research_reports/RR600/RR610/RAND_RR610.pdf

3. Case Studies

To determine how vulnerability disclosure works in practice, this section of the report examines in detail the following four case studies of vulnerabilities:

- Heartbleed;[69]
- Sandworm[70];
- Shellshock[71]; and
- POODLE.[72]

These four vulnerabilities were selected because (i) they were classified as being high/severe impact; (ii) they were easy to access and exploit; (iii) they were (consequently) widely distributed; and (iv) they were all reported in 2014. For each of the case studies, the analysis covers the whole lifecycle of the disclosure process.

3.1 Heartbleed (CVE-2014-0160)

3.1.1 Introduction

The vulnerability known to the public as Heartbleed (CVE-2014-0160) provides attackers with the opportunity to 'steal' information which, under normal conditions, would be protected by SSL/TLS encryption. The seriousness of the Heartbleed bug was broadly recognised, especially since the vulnerability was located in the popular OpenSSL cryptographic software library. The main purpose of the SSL/TLS encryption is to provide communication security and privacy over the Internet for applications such as web, email, instant messaging (IM) and some virtual private networks (VPNs). With the Heartbleed bug security was compromised since the bug provided anyone on the Internet the opportunity to read retrieved data from the memory of the systems protected by the vulnerable versions of the OpenSSL software.[73] The vulnerability exists in all default versions of OpenSSL dating back to March 2012.[74] Sapio (2014) describes the criticality of the vulnerability by indicating how there is a lack of reasonable limits on the type of information that can be compromised; the ability for attacks to be automated and distributed, which complicates identification of possible attackers; and the observation that the attack had potentially been 'in

[69] Common Vulnerabilities and Exposures (CVE) identifier CVE-2014-0160 [National Institute of Standards and Technology. 2015a. 'Vulnerability Summary for CVE-2014-0160.' As of 7 October 2105: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-0160]

[70] Common Vulnerabilities and Exposures (CVE) identifier CVE-2014-4114 [National Institute of Standards and Technology. 2015b. 'Vulnerability Summary for CVE-2014-4114.' As of 7 October 2105: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-4114]

[71] Common Vulnerabilities and Exposures (CVE) identifier CVE-2014-6271 [National Institute of Standards and Technology. 2015c. 'Vulnerability Summary for CVE-2014-6271.' As of 7 October 2105: http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-6271]

[72] Common Vulnerabilities and Exposures (CVE) identifier CVE-2014-3566 [National Institute of Standards and Technology. 2015d. 'Vulnerability Summary for CVE-2014-3566.' As of 7 October 2105: http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-3566]

[73] Heartbleed.com. 2014. 'The Heartbleed Bug.' As of 7 October 2015: http://heartbleed.com/

[74] Sapio, Tim. 2014. 'Heartbleed: Examining the Impact.' DarkReading, 10 April. As of 7 October 2015: http://www.darkreading.com/attacks-breaches/heartbleed-examining-the-impact-/d/d-id/1204330

the wild' for two years prior to its discovery and disclosure.[75] According to Sass (2015), the Heartbleed bug affected over half a million websites.[76]

Glyer (2014) writes how less than a week after the public disclosure of the Heartbleed vulnerability, incident responders from Mandiant had already identified successful attacks by targeted threat actors.[77]

3.1.2 Discovery and disclosure

The narrative of Heartbleed is the unification of two separate discoveries. According to the timeline generated by Grubb (2014), Google researcher Neel Mehta discovered the vulnerability at the latest on 21 March 2014. On that day, two Google employees generated a patch for the vulnerability and distributed it across services and servers around the world. Several days later, on 2 April 2014, a Finnish IT security testing company – Codenomicon – also discovered the vulnerability and reported it to the National Cyber Security Centre (NCSC-FI) in Finland.[78]

Once the NCSC-FI received a report of the vulnerability, the organisation began to verify, analyse and approach the authors of OpenSSL as well as software, operating system and appliance vendors that were potentially affected by the vulnerability. Since the vulnerability had been discovered independently by the two Google employees, the vulnerability had been publicly disclosed before the work of the NCSC-FI had been completed.[79]

On Monday, 7 April 2014, Cloudfare published a blog post entry about Heartbleed. A couple of hours later the company posted a tweet referring to the blog post. Shortly after, Mehta also tweeted about the vulnerability. Codenomicon also tweeted and directly linked to the website they created: heartbleed.com. During the day, the rest of the world was able to read about Heartbleed, and the story about the vulnerability quickly spread through the media.

The disclosure of Heartbleed was a topic of discussion, especially in the media. As described by Grubb (2014), '*Ever since the "Heartbleed" flaw in encryption protocol OpenSSL was made public on April 7 in the US there have been various questions about who knew what and when*.'

JPCert describes how it received information about the vulnerability a few days prior to disclosure, from one of its global counterparts.[80] The JPCert team was about to commence the coordination process with developers when the vulnerability was made public by the OpenSSL team. As indicated in the blog post, this sequence of events showed the JPCert team that even though they received the information in a confidential manner, others may have possessed the same information. This is especially true for open-source products. Uchiyama (2014) writes, '*The experiences that I had over this past year have shown me that not only are more and more people looking for vulnerabilities, but this information is moving around at such high speeds to a variety of parties, where in some cases, have no idea that another particular group has that information*.'[81]

---

[75] Sapio, Tim, 2014.
[76] Sass, Jeff. 2015. 'The Role of Static Analysis in Heartbleed.' SANS Institute. As of 7 October 2015: http://www.sans.org/reading-room/whitepapers/threats/role-static-analysis-heartbleed-35752
[77] Glyer, Christopher. 2014. 'Attackers Exploit the Heartbleed OpenSSL Vulnerability to Circumvent Multi-factor Authentication on VPNs.' FireEye, 18 April. As of 7 October 2015: https://www.mandiant.com/blog/attackers-exploit-heartbleed-openssl-vulnerability-circumvent-multifactor-authentication-vpns/#sthash.WpNMsgIj.dpuf
[78] Heartbleed.com. 2014.
[79] Heartbleed.com. 2014.
[80] Uchiyama, Takayuki. 2014. 'Year in Review - Vulnerability Handling and Changing with the Times.' JP CERT, 11 December. As of 7 October 2015: http://blog.jpcert.or.jp/2014/12/year-in-review---vulnerability-handling-and-changing-with-the-times.html
[81] Uchiyama, Takayuki. 2014

For coordination purposes, however, Uchiyama indicates that it is more beneficial if fewer parties are involved in the process because it reduces the probability of disclosure prior to the development of a means to mitigate the vulnerability. It also provides the developer with more control over the matter.

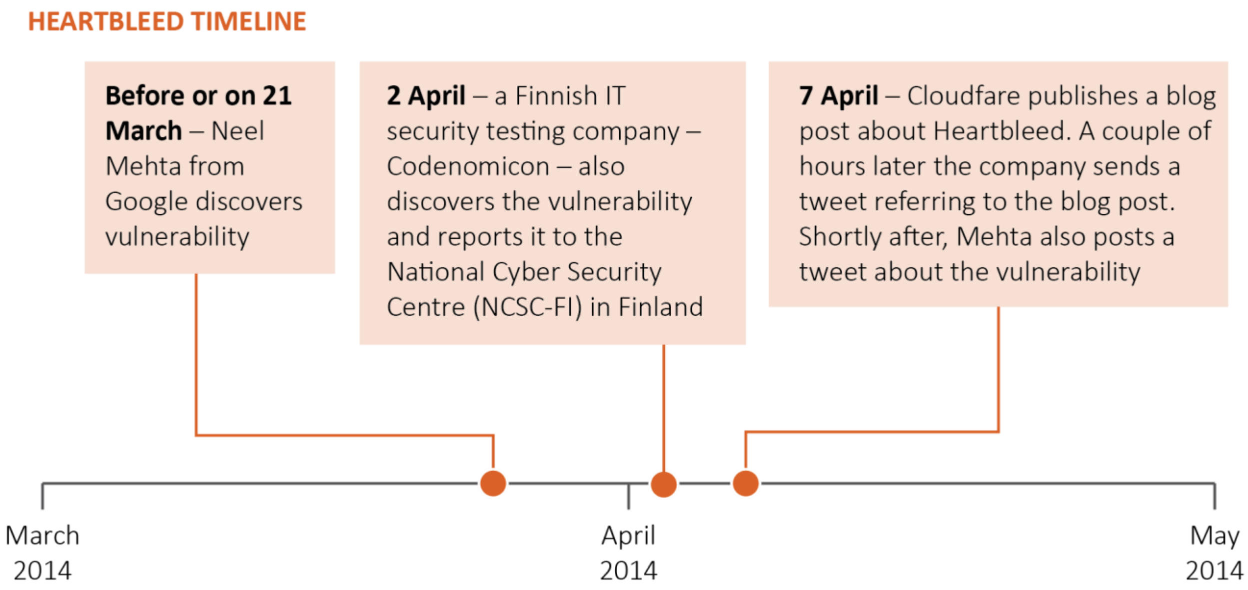


Figure 15: Timeline for the discovery and disclosure of Heartbleed

3.1.3 Aftermath of disclosure

The media attention garnered by the Heartbleed bug in 2014 caused quite a stir. In his blog post on Heartbleed, Kaminsky begins by stating '*we need to take Matthew Green's advice, start getting serious about figuring out what software has become Critical Infrastructure to the global economy, and dedicating genuine resources to supporting that code. It took three years to find Heartbleed. We have to move towards a model of No More Accidental Finds*.'[82] ENISA published a flash note called the Heartbleed vulnerability a 'Wake-up call' [83] about cyber security, and in particular about existing critical vulnerabilities which are dormant.

Another point of discussion in the aftermath of the Heartbleed disclosure was the question revolving around the knowledge of, in particular, the National Security Agency (NSA). Michael Daniel, Special Assistant to the President and the Cybersecurity Coordinator, described in a blog on the White House website how the US government had no knowledge of the vulnerability prior to its disclosure.[84] This was a primary topic of discussion shortly after the disclosure of the vulnerability when reports in the media revealed allegations that the NSA was not only aware of the vulnerability but exploited the vulnerability for its own purposes. The Electronic Frontier Foundation (EFF) filed a Freedom of Information Act (FOIA) suit against the NSA

[82] Kaminsky, Dan. 2014. 'Be Still My Breaking Heart.' Dan Kaminsky's Blog, 10 April. As of 7 October 2015: http://dankaminsky.com/2014/04/10/heartbleed/
[83] https://www.enisa.europa.eu/publications/flash-notes/flash-note-heartbleed-a-wake-up-call
[84] Daniel, Michael. 2014. 'Heartbleed: Understanding When We Disclose Cyber Vulnerabilities.' The White House Blog, 28April. As of 7 October 2015: https://www.whitehouse.gov/blog/2014/04/28/heartbleed-understanding-when-we-disclose-cyber-vulnerabilities

requesting transparency about the decision-making process with respect to vulnerability disclosure by the American government, the discussions are on-going.[85]

Other points of discussion concerning the disclosure involved the role of Google. The company, which was the primary discoverer according to the timeline publicly available, was labelled as 'selfish' by some IT security experts.[86] According to these experts, Google 'waited' for a long time to tell the OpenSSL team and also played 'favourites' by being selective in which private corporations it told about the vulnerability.[87] An interviewee described how it mainly involved a 'circle of friends.' While there is a lack of clarity about the exact date Google researchers discovered the vulnerability, based on publicly information which also includes Google sources reporting to the media, the vulnerability was discovered by Google researchers sometime between 1 March and 21 March, but not reported to OpenSSL until 1 April 2014.[88] This was at the very least eleven days later. This appears to be at odds with the official Google policy as described in one of its blogposts: 'We always report these cases to the affected vendor immediately, and we work closely with them to drive the issue to resolution.'[89]

3.1.3.1 Impact

With such a high-profile vulnerability as Heartbleed, the main question after its disclosure became, 'what is its impact?' IBM security systems characterised Heartbleed as 'one of the most widespread and impactful security vulnerabilities of all time.'[90] Despite the almost instant availability of a solution for Heartbleed, the vulnerability remained a potential danger for users because of the 'ubiquity of OpenSSL.'[91] Any delay in the instalment of the patch meant users remained vulnerable. IBM as a result explains how one-day attacks can be just as dangerous as zero-day vulnerabilities.[92] Users, however, are also dependent on third-party software vendors to find and resolve the vulnerability.[93] Despite the popularity roaming around zero-days, Ablon et al. (2014) write: '*Zero-days are not as prevalent as some might advertise, but they are bought and sold on the black market, if one knows where to find them. What is more prevalent on the black market are "half-days" (or, as one expert calls them, "1-days" or "2-days"), where the software creator may know of the vulnerability and a patch may be available, but few users are aware and implementing those patches*.'[94]

According to Donohue (2014), IBM began witnessing attacks targeting the Heartbleed vulnerability on the same day that the exploit Proof of Concept (PoC) emerged. On 8 April 2014, Jared Stafford published the PoC for a Heartbleed exploit.[95] The PoC was quickly modified and improved.[96] Donohue continues by

[85] Electronic Foundation Frontier (EFF). 2014. 'EFF Sues NSA, Director of National Intelligence for Zero Day Disclosure Process.' EFF, 1 July. https://www.eff.org/press/releases/eff-sues-nsa-director-national-intelligence-zero-day-disclosure-process
[86] Grubb, Ben. 2014. 'Google accused of being selfish and playing favourites over Heartbleed security bug disclosure.' *The Age*, 19 April. As of 7 October 2015: http://www.theage.com.au/it-pro/security-it/google-accused-of-being-selfish-and-playing-favourites-over-heartbleed-security-bug-disclosure-20140418-zqvvk.html
[87] Grubb, Ben. 2014.
[88] The study team did not speak to Google specifically about this vulnerability and its disclosure process.
[89] Evans, Chris & Hintz, Drew, 2013. 'Disclosure timeline for vulnerabilities under active attack.' Google Online Security Blog, 29 May. As of 7 October 2015: http://googleonlinesecurity.blogspot.com.au/2013/05/disclosure-timeline-for-vulnerabilities.html
[90] Donohue, Brian. 2014. 'IBM: Heartbleed Attacks Thousands of Servers Daily.' Threat Post, 27 August. As of 7 October 2015: https://threatpost.com/ibm-heartbleed-attacks-thousands-of-servers-daily/107936#sthash.9qB6TNdR.dpuf
[91] Donohue, Brian. 2014.
[92] A zero-day vulnerability is a vulnerability that is unknown to the vendor and for which as a result no solution is available.
[93] Donohue, Brian. 2014.
[94] Ablon et al. 2014
[95] Stafford, Jared. 2014. 'http://s3.jspenguin.org/ssltest.py.' Pastebin, 8 April. As of 7 October 2015 : http://pastebin.com/WmxzjkXJ
[96] Baggett, Mark. 2014. 'SANS Python Pen Testers | Exploit Heartbleed Vulnerabilities | SEC573.' SANS Penetration Testing, 16 April. As of 8 October 2015 : http://pen-testing.sans.org/blog/2014/04/16/sans-python-pen-testers-exploit-heartbleed-vulnerabilities-sec573-2

describing how IBM indicated that the highest volume of attacks occurred on 15 April, which was a week after the disclosure. At that time, there were more than 300,000 attacks targeting IBM Managed Security Services (MSS) clients in one day. After 22 April, attacks slowed down, according to IBM; yet, the company also indicated how nearly half of all affected systems remain unpatched and that it sees some 7,000 attacks within MSS each day.[97] The Pew Research Center surveyed 1,501 adults living in the United States about Heartbleed and reported how 39% indicated they had changed their passwords or cancelled accounts, but that 6% believed their personal information had been compromised.[98]

Approximately a month after the disclosure, several sources reported that half of the servers remained unpatched and subsequently vulnerable.[99]

3.1.3.2 Looking ahead

In March 2015, Krebs reported on a 'security makeover' introduced by OpenSSL to fix a number of security defects.[100] The disclosure of these updates is particularly relevant in light of the overarching disclosure debate. One of the founding partners of OpenSSL, Steve Marquess, indicated that the information would be shared only with major operating system vendors prior to its public release. As quoted in Krebs, Marquess describes how 'We'd like to let everyone know so they can be prepared and so forth, but we have been slowly driven to a pretty brutal policy of no [advance] disclosure.'[101] This brutal policy Marquess refers to links to a blog post he published after the Heartbleed bug reports and the reactions about the less than ideal manner of disclosure. In this blog post Marquess details the difficulty encountered by the OpenSSL team to do its work based on limited financial support. As he writes in his opening statement, 'As has been well reported in the news of late, the OpenSSL Software Foundation (OSF) is a legal entity created to hustle money in support of OpenSSL. By "hustle" I mean exactly that: raising revenue by any and all means.'[102]

3.2 Sandworm (CVE-2014-4114)

3.2.1 Introduction

In early 2014, a group initially dubbed "Quedagh" grabbed the attention of researchers at F-Secure – an online security and privacy company based in Helsinki, Finland – and ESET – an IT security company headquartered in Bratislava, Slovakia. Quedagh's *modus operandi* was unique as the group utilised a modified BlackEnergy trojan, usually connected to cybercriminal activities such as DDoS attacks, spam distribution, and bank fraud, to infiltrate specific government and private businesses in Ukraine and Poland for the purpose of network discovery, data collection, and remote code execution. According to F-Secure, the use of this modified BlackEnergy trojan (now known as Black Energy 3 or lite) for politically-oriented attacks was particularly intriguing as it provided a '*greater measure of plausible deniability than is afforded by a custom-made piece of code*.'[103] The Quedagh campaign was indeed very sophisticated as it drew on

[97] Donohue, Brian. 2014.
[98] Rainie, Lee & Maeve Duggan. 2014. 'Heartbleed's Impact.' Pew Research Centre, 30 April. As of 8 October 2015: http://www.pewinternet.org/2014/04/30/heartbleeds-impact/
[99] Ring, Tim. 2014. 'Tens of thousands of servers *still* vulnerable to Heartbleed.' *SC Magazine*, 9 May. As of 8 October 2015 : http://www.scmagazineuk.com/tens-of-thousands-of-servers-still-vulnerable-to-heartbleed/article/346268/
[100] Krebs On Security. n.d. 'Posts Tagged: Heartbleed.' As of 8 October 2015: http://krebsonsecurity.com/tag/heartbleed/; Caswell, Matt. '[openssl-announce] Forthcoming OpenSSL releases.' Message to openssl-announce mailing list. 16 March 2015. Email: http://marc.info/?l=openssl-announce&m=142653572011212&w=2
[101] Krebs On Security. n.d.
[102] Marquess, Steve. 2014. 'Of Money, Responsibility, and Pride.' SPEEDS AND FEEDS, 12 April. As of 8 October 2015: http://veridicalsystems.com/blog/of-money-responsibility-and-pride/
[103] F-Secure. 2014. 'Blackenergy & Quedagh – The Convergence of crimeware and APT attacks.' As of 8 October 2015: https://www.f-secure.com/documents/996508/1030745/blackenergy_whitepaper.pdf

'*technical infection methods through the exploitation of software vulnerabilities (CVE 2014-1761), social engineering through spear-phishing emails and decoy documents, or a combination of both*.'[104]

3.2.2 Discovery and disclosure

On 3 September 3 2014, iSIGHT Partners discovered the use of a previously unknown Windows zero-day exploit (CVE-2014-4114) by a group they dubbed 'Sandworm.'[105] The group was dubbed Sandworm because they used 'encoded references to the classic science fiction series *Dune* in command and control URLs and various Malware samples.' This particular zero-day exploit was very dangerous, to the extent that it affected all supported versions of Microsoft Windows[106] and was purportedly specifically harnessed to target NATO, the Ukrainian and Western governments, energy sector firms in Poland (in combination with CVE 2014-3906[107]), telecommunication companies in France, and at least one think tank in the United States. No other use of this zero-day exploit is known outside the Sandworm context.

iSIGHT Partners contacted the parties and clients they were able to identify, and started collaborating with Microsoft on 5 September 2014 to deliver the technical analysis of the exploit concerned and the malware used in the attacks. Both companies furthermore coordinated the tracking of the group's activities by monitoring Sandworm's targeting behaviour and the broader use of this zero-day exploit in the wild.[108]

As a result of this collaboration, previously unconnected pieces of information began to reveal a comprehensive picture of what is now known as the Sandworm campaign. On the basis of overlapping infrastructure, use of traditional crimeware, and unique references to *Dune*, iSiGHT Partners were able to trace the genesis of Sandworm to the year 2009. They were also able to connect the group to various other attacks on NATO in 2013, targeted attacks on the 2014 GlobeSec Meeting in Bratislava, and the 'Quedagh' operation.

3.2.2.1 Zero-day (CVE-2014-4114)

The Windows zero-day exploit used by Sandworm exposed a dangerous vulnerability across all supported Microsoft Windows systems. According to TrendMicro the vulnerability was located in the PACKAGER.DLL file which is part of the Windows Object Linking and Embedding property (OLE).[109] OLE's basic functionality is to make content, such as text, images, and programme files created in one programme available in another. Sandworm, however, used embedded OLE content to weaponise Microsoft Office documents such as PowerPoint files. Once opened, the PowerPoint presentation would run in animation mode and execute OLE objects without prompting additional user action.

The embedded OLE container essentially included a URL that linked to an .INF and a .GIF file (which was actually a renamed .EXE file) which were automatically downloaded once the PowerPoint file was opened. The .INF file was used to automatically rename the .GIF into an .EXE file, and created a '*runonce'* entry in the system registry which would execute the .EXE file once the system rebooted. The then installed BlackEnergy malware created a backdoor which enabled remote access for the attacker.

[104] F-Secure. 2014.
[105] Ward, Stephen. 2014. 'iSIGHT discovers zero-day vulnerability CVE-2014-4114 used in Russian cyber-espionage.' iSIGHT Partners Blog, 14 October. As of 8 October 2015: http://www.isightpartners.com/2014/10/cve-2014-4114/
[106] The vulnerability does not appear to affect Windows XP
[107] Ward, Stephen, 2014
[108] Ward, Stephen, 2014
[109] Wu, Weimin. 2014. 'An Analysis of Windows Zero-Day Vulnerability 'CVE 2014-4114' aka Sandworm.' Trendlabs Security Intelligence Blog, 14 October. As of 8 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/an-analysis-of-windows-zero-day-vulnerability-cve-2014-4114-aka-sandworm/

Given that knowledge of the zero-day exploit was limited and that its use was confined to the Sandworm campaign, iSIGHT and Microsoft purposely delayed public disclosure to coincide with Microsoft's Patch Tuesday.[110] The logic for withholding information from the public was based on the argument that a patch inclusion in the monthly automatic Windows update would limit exposure to this wide-reaching and severe Windows vulnerability, and also minimise the potential for any copy-cat exploit creation.[111]

Apart from the Microsoft Patch (MS14-060) to close down the vulnerability, iSIGHT also released a set of workarounds in case automatic Windows updates were disabled on systems for various reasons. Users could choose between disabling the WebClient Sevice,[112] blocking TCP ports 139 and 445,[113] or blocking launching executables via setup information files.[114] Symantec additionally advised businesses and consumers to 'exercise caution when opening email attachments, particularly from unknown sources.'[115]

The National Institute for Standards and Technology (NIST) ranked the Windows OLE Remote Execution Vulnerability at a 9.3 (High) base score.[116] However, as Ross Barrett, Senior Manager at Rapid7 noted, Microsoft only called the issue 'important' with a patching priority two, which is one level down from urgent patching and the most severe rating.[117] Microsoft's decision was primarily based on the fact that the exploit would require a user to click on a file. Qualys on the other hand ranked the exploit as severe, explaining that 'it is pretty easy to trick a single person into clicking on a file.'[118]

3.2.3 Aftermath of disclosure

3.2.3.1 Zero-day Redux (CVE 2014-6352)

On 21 October 2014, Haifei Li at McAfee noted in a blog post that Microsoft's official patch was not robust enough to close down the entire zero-day vulnerability exposed by Sandworm[119].

Indeed, in cooperation with James Forshaw at Google's Project Zero, Li reported the issue to Microsoft's Security Response Center on 17 October 2014 after successfully developing a proof-of-concept for the still existing vulnerability gap (CVE 2014-6352). Coinciding with Li's blog post, Microsoft released a temporary 'fix it' patch, but no further details were provided because no permanent patch was yet available to the public. On 11 November 2014, Microsoft's Security Advisory for CVE 2014-6352 was finally updated. Apparently the zero-day vulnerability also extended to OLE objects in webpages when using Internet

[110] 'Microsoft's 'Patch Tuesday' generally occurs on the second Tuesday of each month.
[111] Ward, Stephen. 2014.
[112] Impact: Any service depending on Web Client service will not start
[113] Ports 139 and 445 are used for additional services such as Common Internet File Systems, DNS Administration, NetBT service sessions, printer sharing sessions and more.
[114] Impact: Applications that rely on the use of .INF files to execute an installer application may not automatically execute.
[115] Symantec Security Response. 2014. 'Sandworm Windows zero-day vulnerability being actively exploited in targeted attacks.' Symantec Security Response Blog, 14 October. As of 8 October 2015: http://www.symantec.com/connect/blogs/sandworm-windows-zero-day-vulnerability-being-actively-exploited-targeted-attacks
[116] National Institute of Standards and Technology. 2014b. 'National Cyber Awareness System – Vulnerability Summary for CVE-2014- 4114.' As of 8 October 2015: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-4114
[117] Barrett, Ross. 2014. 'Microsoft patches SandWorm 0-day.' Net-Security, 14 October. As of 8 October 2015: http://www.net-security.org/secworld.php?id=17492
[118] Jackson, Joab. 2014. 'Microsoft Patch Tuesday tackles three critical vulnerabilities, including 'Sandworm'.' PC World, 14 October. As of 8 October 2015: http://www.pcworld.com/article/2833852/microsoft-patch-tuesday-tackles-three-critical-vulnerabilities.html

[119] https://blogs.mcafee.com/mcafee-labs/new-exploit-sandworm-zero-day-bypass-official-patch/

Explorer. In essence if a user was logged on with full administrative rights, 'an attacker could install programs; view, change, or delete data; or create new accounts with full user rights.'[120]

On 16 October 2014, TrendMicro uncovered the fact that Sandworm targeted Supervisory Control and Data Acquisition (SCADA)-centric systems which used the GE Intelligent Platform CIMPLICTY HMI solution suite. Further investigation by iSIGHT revealed that WinCC and the Siemens HMI were also targeted with a system-specific BlackEnergy payload. Although TrendMicro clarified that they 'found no indications that this malware is actually manipulating physical SCADA systems or their resultant data',[121] the intrusions might have been reconnaissance work for a future attack.[122] According to Wolfgang Kandek, Chief Technology Officer at Qualys, Sandworm was a reminder to system administrators to ensure that user permissions are set correctly.[123]

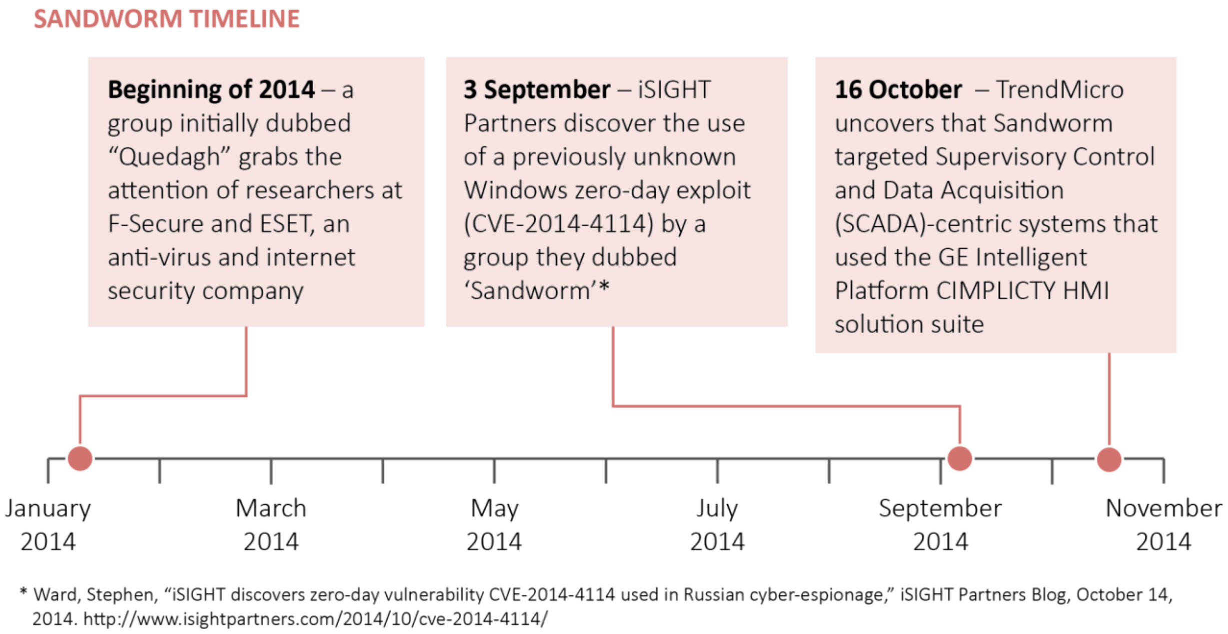


Figure 16: Timeline for the discovery and disclosure of Sandworm

3.3 Shellshock (CVE-2014-6271)

3.3.1 Introduction

CVE-2014-6271[124] was a critical remote code execution vulnerability that was discovered in the widely used GNU Bourne-Again Shell (or Bash) in September 2014. The Bash command shell is a free programme installed in Unix-based computer systems to allow users to execute interactive command scripts (e.g. various flavours

---

[120] Microsoft Security Bulletin MS14-064. 2014. 'Vulnerabilities in Windows OLE Could Allow Remote Code Execution (3011443).' Security TechCenter, 11 November. As of 8 October 2015: https://technet.microsoft.com/library/security/MS14-064
[121] Wilhoit, Kyle & Gogolinski, Jim. 2014. 'Sandworm to Blacken – The SCADA Connection.' Trendlabs Security Intelligence Blog, 16 October. As of 8 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/sandworm-to-blacken-the-scada-connection/
[122] Hultquist, John. 2014. 'Sandworm Team – Targeting SCADA Systems.' iSIGHT Partners Blog, 21 October. As of 8 October 2015: http://www.isightpartners.com/2014/10/sandworm-team-targeting-scada-systems/
[123] Jackson, Joab, 2014.
[124] National Institute of Standards and Technology. 2015c.

of the Linux operating system, MacOS X and even Windows-based applications such as Cygwin[125] rely on Bash).[126] In addition, certain programmes and scripts (e.g. Secure Shell and Common Gateway Interface) permit Bash to run in the background on systems thereby allowing attackers to potentially take advantage of the vulnerability remotely.[127] 'Shellshock', as the vulnerability has come to be known, enabled attackers to acquire unauthorised and unhindered access to millions of networked computer systems and devices around the world, and essentially 'tell' the affected systems what to do. Shellshock thus represented a highly significant security concern when it was discovered and subsequently disclosed.

3.3.1.1 Background

The Bash software was written in 1987 by Brian Fox, a computer programmer, who also maintained the software for the next five years. In 1992, the maintenance of Bash was handed over to Chet Ramey, currently a Senior Technology Architect at Case Western Reserve University in the United States.[128] Ramey has been voluntarily maintaining the software as a 'hobby' since 1992. The bug appears to have been accidentally introduced in August 1989 by Fox during an update of Bash.[129]

The types of systems which were potentially vulnerable as a consequence of the bug included devices such as standard computers, servers and routers. Even industrial products were at serious risk of compromise, as indicated by advisory alerts issued by large corporations.[130] As a result of lying undetected for almost 25 years, some analysts projected that Bash was built into as many as 70% of computer systems that were connected to the Internet, running into the hundreds of millions.[131] Indeed, one estimate noted that around 50% of all web servers around the world are Unix-based, resulting in approximately 500 million of these systems that were potentially at risk.[132] In summary, Shellshock 'made benign-seeming server requests into a full command-and-control situation',[133] and it was 'assumed that most server-based architectures [could have been] affected.'[134]

3.3.2 Discovery and disclosure

The vulnerability in Bash was discovered on 12 September 2014 by Stephane Chazelas, an IT Manager and Unix/open source enthusiast based in Edinburgh, United Kingdom. Chazelas detected the bug after

---

[125] Cygwin provides a Unix-like environment for Microsoft Windows based systems.
[126] For example, as the command language interpreter on Unix-based systems, Bash allows users to carry out editing, completion, integer arithmetic, string-related, and input-output-related tasks. [Ramey, Chet. n.d. 'BASH - The Bourne-Again Shell.' As of 1 May 2015: http://tiswww.case.edu/php/chet/bash/bash-intro.html]
[127] Trend Micro. 2014., 'Shellshock: A Technical Report.' As of 9 October 2015: http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-shellshock.pdf
[128] Scott, Rob. 2014. 'Security Experts Warn of Potentially Dangerous Shellshock Bug .' ETCentric, 26 September. As of 12 October 2015: http://www.etcentric.org/security-experts-warn-of-potentially-dangerous-shellshock-bug
[129] Wheeler, David A. 2015. 'Shellshock.' Dweeler.com, 13 February. As of 26 May 2015: http://www.dwheeler.com/essays/shellshock.html#timeline
[130] Kovacs, Eduard. 2014. 'Several Siemens Industrial Products Affected by ShellShock Bug.' Security Week, 8 October. As of 28 April 2015: http://www.securityweek.com/several-siemens-industrial-products-affected-shellshock-bug
[131] Perlroth, Nicole. 2014. 'Security Experts Expect 'Shellshock' Software Bug in Bash to Be Significant.' New York Times, 25 September. As of 29 April 2015: http://www.nytimes.com/2014/09/26/technology/security-experts-expect-shellshock-software-bug-to-be-significant.html?_r=3
[132] Francis, Hannah. 2014. 'Shellshock: The latest security superbug explained.' The Sydney Morning Herald, 27 September. As of 29 April 2015: http://www.smh.com.au/digital-life/consumer-security/shellshock-the-latest-security-superbug-explained-20140927-10mcfx.html
[133] *The Economist*. 2014. 'Ghosts in the machine language.' *The Economist*, 24 October. As of 21 April 2015: http://www.economist.com/news/science-and-technology/21627868-latest-hacks-and-exploits-result-benign-neglect-and-wont-be-last-ghosts-machine
[134] CERT-UK. 2014. 'UPDATE: Bash Vulnerability AKA SHELLSHOCK.' CERT-UK, 9 October. As of 26 May 2015: https://www.cert.gov.uk/resources/advisories/update-bash-vulnerability-aka-shellshock/

discovering a similar issue in another system (GNU libc) a few months previously.[135] He immediately reported the vulnerability to Chet Ramey (who maintains Bash) and a few others (e.g. security contacts at major Unix-based operating system vendors like Debian, Red Hat and Ubuntu).[136] Specifically, Chazelas mentions that he reported '*details of the bug…with a big fat warning that it was very serious and [was] not to be disclosed*.'[137] Although Chazelas notes that '*he was out of the loop after the 19th [of September]*'. With regard to the public disclosure of the vulnerability, he recalls, '*a release schedule with public disclosure on the 24th at 14:00 UTC and early notification to other Unix and Linux vendors on the 22nd and select infrastructure provider notification (such as CDNs including Microsoft) on the 23rd*'[138]

The vulnerability was assigned the Common Vulnerabilities and Exposures (CVE)[139] identifier CVE-2014-6271. Ramey and his team developed a patch and then contacted a selected group of major vendors and distributors who were most likely to be directly influenced by the discovery of the security vulnerability. 'Shellshock', a name proposed by Andreas Lindh on Twitter,[140] was disclosed to the public on 24 September 2014. Figure 17 presents a screenshot of one of the first disclosures of Shellshock that was made on a security mailing list.

CVE-2014-6271: remote code execution through bash

From: Florian Weimer <fw () deneb enyo de>
Date: Wed, 24 Sep 2014 16:05:51 +0200

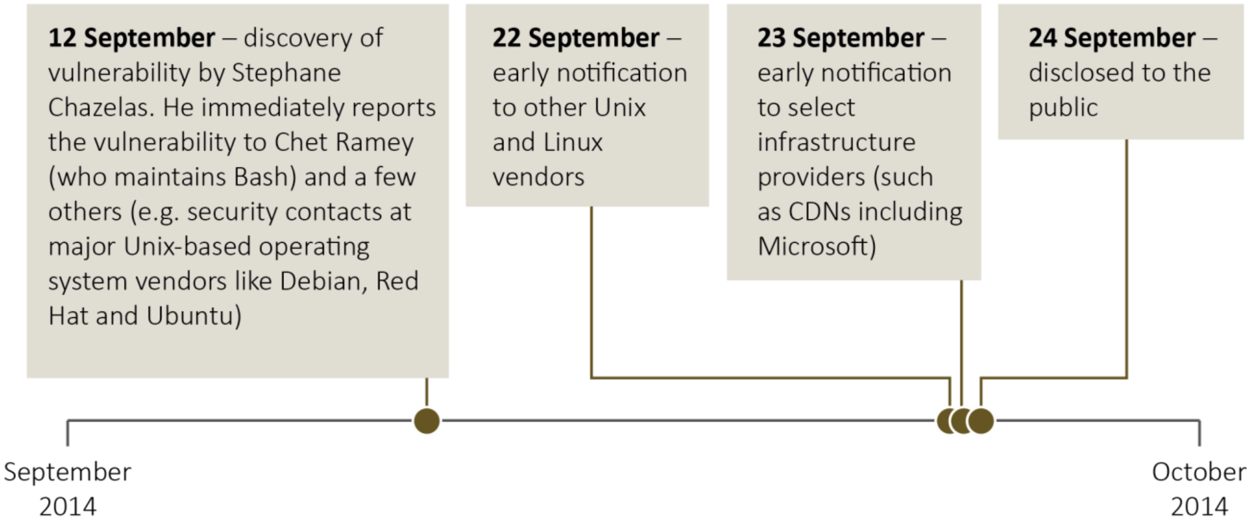
```
Stephane Chazelas discovered a vulnerability in bash, related to how
environment variables are processed: trailing code in function
definitions was executed, independent of the variable name.

In many common configurations, this vulnerability is exploitable over
the network.

Chet Ramey, the GNU bash upstream maintainer, will soon release
official upstream patches.
```

Figure 17: Screenshot of one of the first disclosures of Shellshock made on a security mailing list[141]

[135] Chazelas, Stephane. 2014, 8 October. 'How *DID* you find Shellshock?.' Message to David A Wheeler. Email. As of 26 May 2015: http://www.openwall.com/lists/oss-security/2014/10/08/17
[136] Grubb, Ben. 2014. 'Stephane Chazelas: the man who found the web's 'most dangerous' internet security bug.' 27 September, The Age. As of 26 October 2015: http://www.theage.com.au/it-pro/security-it/stephane-chazelas-the-man-who-found-the-webs-most-dangerous-internet-security-bug-20140926-10mixr
[137] Chazelas, Stephane. 2014, 3 October. 'RE: Shellshock Timeline.' Message to David A Wheeler. Email. As of 27 May 2015: http://seclists.org/oss-sec/2014/q4/92
[138] Chazelas, Stephane. 2014, 3 October.
[139] CVE. 2015.
[140] Lindh had actually proposed the name 'Shell Shock.' Lindh, Andreas (addelindh). '@markstanislav Shell schock.' 24 September 2014, 9:42. As of 27 May 2015: https://mobile.twitter.com/addelindh/status/514817121101283328
[141] Weimer, Florian. 2014, 24 September. 'CVE-2014-6271: remote code execution through bash.' Email. As of 15 October 2015: http://seclists.org/oss-sec/2014/q3/649

The original fix proposed by Ramey and his team, however, did not completely resolve the issue as it was soon discovered by security researchers that there were still glaring security holes in Bash.[142,143] Within days of the original disclosure, a series of other related critical vulnerabilities were discovered and each assigned its own CVE identifier (CVE-2014-6277,[144] CVE-2014-6278,[145] CVE-2014-7169,[146] CVE-2014-7186[147] and CVE-2014-7187[148]). A number of further security advisories and patches were rapidly issued to address these additional vulnerabilities.[149] These included comprehensive security updates from major Unix-based software vendors such as Red Hat,[150] [151] Fedora[152], SUSE[153], Canonical (Ubuntu)[154] and Apple.[155]

SHELLSHOCK TIMELINE



[142] Vaughan-Nichols, Steven J. 2015. 'Shellshock: Better 'bash' patches now available.' ZD Net, 27 September. As of 27 May 2015: http://www.zdnet.com/article/shellshock-better-bash-patches-now-available/
[143] Wheeler, David A. 2015.
[144] Discovered by Michal Zalewski (Google). Mimoso, Michael. 2014. 'Researcher takes wraps off two undisclosed shellshock vulnerabilities in Bash.' Threatpost, 3 October. As of 12 October 2015: https://threatpost.com/researcher-takes-wraps-off-two-undisclosed-shellshock-vulnerabilities-in-bash/108674/
[145] Discovered by Michal Zalewski (Google). Mimoso, 2014.
[146] Discovered by Tavis Ormandy (Google).Bisht, Virendra & William Gamazo Sanchez. 2014. 'Shellshock vulnerabilities proliferate, affect more protocols.' Trend Micro, 2 October. As of 12 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/shellshock-vulnerabilities-proliferate-affect-more-protocols/
[147] Discovered independently by Florian Weimer (Red Hat) and Todd Sabin. Bisht & Sanchez 2014.
[148] Discovered by Florian Weimer (Red Hat). Bisht & Sanchez 2014.
[149] Vaughan-Nichols, Steven J. 2015.
[150] Red Hat. 2014a, 26 September. 'Important: bash security update.' As of 27 May 2015: https://rhn.redhat.com/errata/RHSA-2014-1306.html
[151] Red Hat. 2014b, 2 October. 'Bash Code Injection Vulnerability via Specially Crafted Environment Variables (CVE-2014-6271, CVE-2014-7169).' As of 27 May 2015: https://access.redhat.com/articles/1200223
[152] Fedora Update Notification. 2014. '[SECURITY] Fedora 21 Update: bash-4.3.25-2.fc21.' As of 27 May 2015: https://lists.fedoraproject.org/pipermail/package-announce/2014-September/139129.html
[153] Open SUSE Security.n.d. 'Mailinglist Archive: opensuse-security-announce (44 mails).' As of 27 May 2015: http://lists.opensuse.org/opensuse-security-announce/2014-09/msg00042.html
[154] Ubuntu. 2014, 27 September. 'USN-2364-1: Bash vulnerabilities.' As of 27 May 2015: http://www.ubuntu.com/usn/usn-2364-1/
[155] Apple. 2014. 'OS X bash Update 1.0 – OS X Mavericks.' As of 27 May 2015: https://support.apple.com/kb/DL1769?viewlocale=en_US&locale=en_US

Figure 18: Timeline for the discovery and disclosure of Shellshock

3.3.3 Aftermath of disclosure

Immediately after disclosure of the vulnerability, Shellshock was declared to be even more dangerous and a far bigger threat than the Heartbleed bug which had been discovered only a few months earlier. Shellshock allowed unauthorised access to 'infected' machines, thereby enabling attackers to take control of entire machines.[156] Heartbleed, on the other hand, largely permitted hackers to 'spy' on infected systems (for example, steal passwords). To quote a security researcher, '*Whereas something like Heartbleed was all about sniffing what was going on, this was about giving you direct access to the system…. The door's wide open*.'[157] Notably, Shellshock was classified as high impact and low complexity in the NVD's[158] CVSS, and was given the maximum ratings (10) on the 10-point CVSS impact and exploitability scales.[159] By permitting the remote execution of arbitrary code on affected systems, the critical vulnerability allowed (i) unauthorised disclosure of information; (ii) unauthorised modification; and (iii) disruption of service.[160]

As the vulnerability lay undiscovered for more than 25 years, millions of computer systems across the globe had incorporated and built on top of the 'flawed' Bash code, thus explaining the widespread 'fear factor' associated with its sudden disclosure in September 2014. The immediate aftermath of the disclosure saw widespread media coverage around the world warning of the looming impact of the vulnerability on potentially millions of devices.

Immediately after disclosure, some security companies noted parallel increases in traffic related to the progression of the validating process (by security researchers), testing of environments (by organisations), and targeted attacks to take advantage of the vulnerability.[161] As Tran (2014) reported, 'Within hours of the release of this bug to the general public, attackers reportedly exploited this vulnerability to create botnets on compromised computers to perform DDos (distributed denial-of-service) and vulnerability scanning.'[162] For example, within one day of the disclosure, some attackers had taken advantage of the vulnerability and had built a botnet (called 'wobpot') that targeted systems at the US Department of Defense and Akamai, a leading US-based content delivery network provider and cloud platform.[163] Annex D lists sample text from some of the advisories and alerts that were issued by various organisations around the world within hours of the disclosure of the vulnerability.

3.4 POODLE (CVE-2014-3566)

3.4.1 Discovery and disclosure

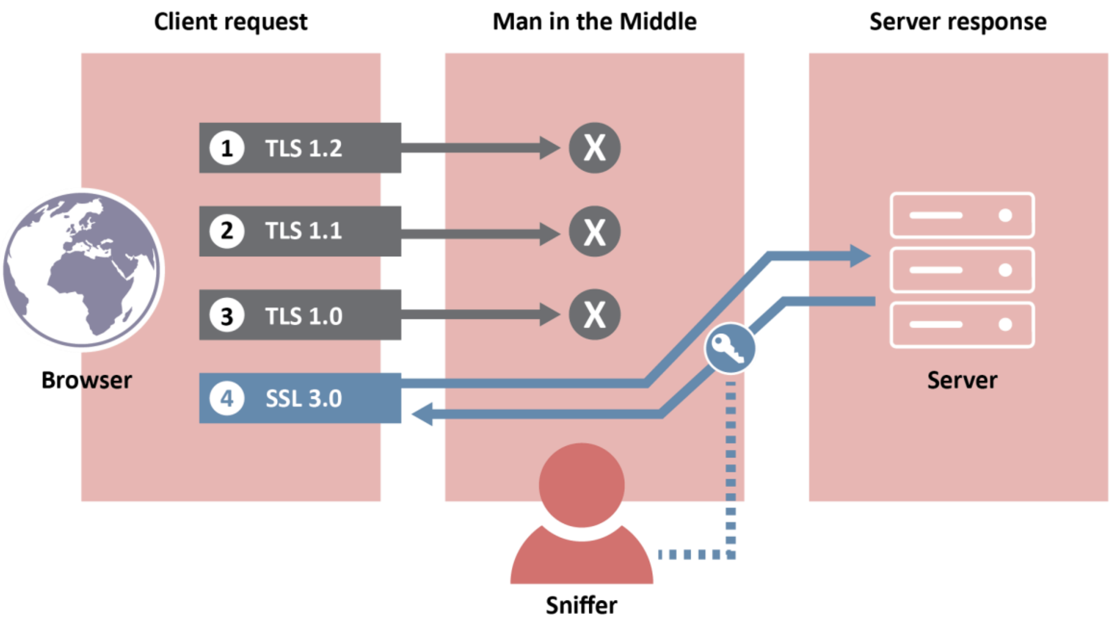
On 14 October 2014, Google Security team member Bodo Möller, in cooperation with Thai Duong and Krzysztof Kotowicz, published a vulnerability connected to the 18-year-old Security Socket Layer (SSL) 3.0 protocol which they dubbed '*Padding Oracle On Downgraded Legacy Encryption [POODLE]*'. While

[156] Kahl, Chad. 2014a. 'The Shellshock BaSH Bug: Vulnerability in BaSH is a Big Deal.' Solutionary, 25 September. As of 29 April 2015: http://www.solutionary.com/resource-center/blog/2014/09/shellshock-vulnerability-in-bash-is-a-big-deal/
[157] Lee, Dave. 2014. 'Shellshock: 'Deadly serious' new vulnerability found.' BBC, 25 September. As of 29 April 2015: http://www.bbc.com/news/technology-29361794
[158] National Institute of Standards and Technology. n.d. – a.
[159] National Institute of Standards and Technology. 2015c.
[160] National Institute of Standards and Technology. 2015c.
[161] Kahl, Chad. 2014b. 'Shellshock: Accelerating The Standard Timeline.' Solutionary, 26 September. As of 5 May 2015: http://www.solutionary.com/resource-center/blog/2014/09/shellshock-accelerating-the-standard-timeline/
[162] Tran, Cindee. 2015. 'Zero-day Attacks in 2014.' App Sec Consulting, 16 January. As of 5 May 2015: https://www.appsecconsulting.com/blog/zero-day-attacks-in-2014
[163] Saarinen, Juha. 2014. 'First Shellshock botnet attacks Akamai, US DoD networks.' IT News, 26 September. As of 26 May 2015: http://www.itnews.com.au/News/396197,first-shellshock-botnet-attacks-akamai-us-dod-networks.aspx

vulnerabilities in SSL 3.0 are usually not considered news anymore, POODLE was a different challenge as it allowed network attackers to downgrade (e.g. force) clients and servers to use SSL 3.0 through continuous connection failures. A subsequent man-in-the-middle (MITM) attack,[164] similar to BEAST[165], then allowed encrypted traffic to be decrypted by the attacker (Figure 19).

For all practical purposes SSL 3.0 has long been deemed an obsolete and insecure protocol[166] which was widely replaced by its successors: Transport Layer Security (TLS) 1.0, TLS 1.1, and TLS 1.2. Indeed, SSL's only systemic purpose today is to allow for backwards compatibility to guarantee interoperability with legacy systems (ex. IE6/XP and older) and for providing a smoother user experience.[167]

The encryption cypher suits in SSL 3.0 are particularly worrisome as SSL 3.0 either uses a RC4 stream cypher or an AES CBC-mode bloc cypher. According to Bodo et al. both cyphers are deemed unsafe, and in contrast to other related forms of attack there was no reasonable workaround for POODLE. As a result Bodo et al. concluded that 'to achieve secure encryption, SSL 3.0 must be avoided entirely.'[168]



[164] Man-in-the-Middle (MITM) attack is an attack where the attacker, unbeknownst to the other parties involved in the communication, relays and possibly alters the communication between two parties who believe they are directly communicating with each other.
[165] BEAST (Browser Exploit Against SSL/TLS), is an exploit that was practically demonstrated by Julian Rizzo and Thai Duoang at the 2011 Ekoparty Security Conference in Buenos Aires, Argentina. BEAST leverages a weakness in the cypher block chaining (CBC) to exploit SSL protocol to decrypt and obtain authentication tokens and cookies from HTTPS requests.
[166] Weith, Loren. 2006. 'Differences between SSLv2, SSLv3, and TLS. Available from: http://www.yaksman.org/~lweith/ssl.pdf
[167] Möller, Bodo, Thai Duong & Kotowicz, Krzysztof. 2014. 'Security Advisory - This POODLE Bites: Exploiting the SSL 3.0 Fallback.' Available from: https://www.openssl.org/~bodo/ssl-poodle.pdf
[168] Möller, Duong & Kotowicz, 2014

Figure 19: Illustration of "Man in the Middle" attack[169]

3.4.2 Aftermath of disclosure

3.4.2.1 Response from the information security community

Google's recommendation was thus fairly straightforward: either disable SSL 3.0 support altogether or use a TLS Fallback Signaling Cipher Suite value which prevents attackers from inducing clients and servers to fall back to SSL 3.0 in the first place. Patching SSL 3.0 would have also been a realistic option, but was quickly dismissed as being neither feasible nor a sufficiently prompt solution to the problem at hand.[170]

Google's servers had been supporting TLS FALLBACK SCSV since February 2014.[171] Subsequent versions of their Chrome browser removed SSL 3.0 completely. Microsoft published a Security Advisory on 14 October and Mozilla offered a 'SSL Version Control Firefox extension' to disable SSL 3.0.[172] Oracle recommended to its customers to permanently disable SSL v3.0. Their quarterly Critical Patch Update disabled SSL 3.0 in the Java Runtime Environment on 20 January. Cisco released a Security Advisory on 15 October while noting that they are 'not aware of any malicious use of the vulnerability.'[173] IBM offered an update on 20 October which disabled SSL 3.0 by default for HIS 7.0 and newer.

For all the attention devoted to the vulnerability, POODLE received a relatively low (4.3 – medium) base score on the NVD's[174] CVSS.[175] The relatively low score underlines that POODLE is indeed tied to specific scenario parameters which limit exposure. First, an attacker would have to be on the same network as the user in order to exploit the vulnerability (or gain ISP-level interception through DNS poisoning).[176] Secondly, the user must be running JavaScript.[177] And thirdly, client and server must support SSL 3.0.[178] Only if these three conditions are met can a network attacker intercept, for example, encrypted HTTPS session cookies to hijack secure web sessions.

POODLE in essence offered defenders five vectors of mitigation which were almost equally effective in closing down the vulnerability: (i) disable SSL 3.0 on the client side; (ii) disable SSL 3.0 on the server side; (iii) disable SSL 3.0 in Java Script; (iv) Install a TLS Fallback on the Client side; and (v) or install a TLS Fallback on the server side.

---

[169] Chang, Ziv. 2014. 'POODLE vulnerability puts online transactions at risk.' TrendLabs Security Intelligence Blog, 15 October. As of 22 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/poodle-vulnerability-puts-online-transactions-at-risk/
[170] Red Hat Product Security. 2014. 'Can SSL 3.0 be fixed? An analysis of the POODLE attack.' Security Blog, 20 October. As of 8 October 2015: https://securityblog.redhat.com/2014/10/20/can-ssl-3-0-be-fixed-an-analysis-of-the-poodle-attack/
[171] Möller, Bodo. 2014. 'This POODLE bites: exploiting the SSL 3.0 fallback.' Google Online Security Blog, 14 October. As of 8 October 2015: http://googleonlinesecurity.blogspot.be/2014/10/this-poodle-bites-exploiting-ssl-30.html
[172] Barnes, Richard. 2014. 'The POODLE Attack and the End of SSL 3.0.' Mozilla Security Blog, 14 October. As of 8 October 2015: https://blog.mozilla.org/security/2014/10/14/the-poodle-attack-and-the-end-of-ssl-3-0/
[173] Cisco Security Advisory. 2014. 'SSL Padding Oracle On Downgraded Legacy Encryption (POODLE) Vulnerability.' As of 6 October 2015: http://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20141015-poodle
[174] National Institute of Standards and Technology. n.d. – a.
[175] National Institute of Standards and Technology. 2015c.
[176] Nolette, Ryan. 2014. 'After Taking a Bite Out of SSL 3.0, This POODLE Needs Some Time in Obedience Class.' Bit9 Blog, 15 October. As of 8 October 2015: https://blog.bit9.com/2014/10/15/after-taking-a-bite-out-of-ssl-3-0-this-poodle-needs-some-time-in-obedience-class/
[177] Nolette, Ryan, 2014.
[178] Geftic, Seth. 2014. 'Patching Poodles and Digging for Sandworms: Why Monitoring Matters.' RSA Security Operations Blog, 17 October. As of 8 October 2015: https://blogs.rsa.com/patching-poodles-digging-sandworms-monitoring-matters/

As a result, Tal Klein, Vice-President at Adallom summarised: 'if Shellshock and Heartbleed were Threat Level 10, then POODLE is more like a 5 or a 6.'[179] Ivan Ristic, Director of Application Security Research at Qualys, joined in the chorus by adding that 'POODLE was not as serious as the previous threats because the attack was quite complicated, requiring hackers to have privileged access to networks.'[180] The seriousness of a vulnerability could have an impact on the manner in which a vulnerability is disclosed and as such is an important characteristic. Along with seriousness, however, as indicated in the quote above, the likelihood that a vulnerability can and will be exploited is a potential influential factor on manner of disclosure.

3.4.2.2 POODLE bites again (CVE-2014-8730)

While patches to fix the POODLE vulnerability were gradually implemented, by either disabling SSL 3.0 (thus relying on TLS to avoid the problem) or by installing a TLS FALLBACK SCSV, Brian Smith (formerly of Mozilla) raised the notion that these fixes do not necessarily resolve the underlying padding problem. On 17 October Brian wrote that he fixed a bug in NSS back in 2010, 'where NSS did not verify all the padding bytes in TLS 1.0 records. Thus, any server that is using a version of NSS released prior to June 2010 is likely vulnerable to POODLE-like attacks even if SSL 3.0 is completely disabled.'[181]

In essence the problem came down to an implementation flaw in the CBC encryption mode. SSL 3.0, for example, under-specifies the content of the CBC padding bytes. Implementations can therefore not check the padding bytes, which open SSL 3.0 up to an Oracle attack. TLS 1.0 specifications on the other hand do not specify that implementations must check the padding. That requirement was only added in subsequent TLS versions. 'Thus, an implementation could completely conform to TLS 1.0 but still [be] vulnerable to POODLE.'[182] To make the problem worse, while TLS 1.1 and 1.2 implementations must check the padding, it is not necessarily the case 'that otherwise-working implementations actually conform to that requirement, and there's no way for the client to check that in a reliable, high-performance, and accurate way.'[183]

Adam Langley at Google summarised the issue on 8 December by stating that 'if an SSLv3 decoding function was used with TLS, then the POODLE attack would work, even against TLS connections.'[184] Langley therefore concluded that 'everything less than TLS 1.2 with an AEAD cipher suite is cryptographically broken.'[185]

Independently from Brian Smith, the problem was also investigated by Yngve Nysaeter Pettersen at Vivaldi. On December 9, Pettersen noted that 3.6% of servers are vulnerable to a POODLE-style attack even if they disable SSL 3.0. Whether this concerns all global servers, however, remains unreported. The problem furthermore compounds with 4.24% of TLS 1.2 servers having the same vulnerability, 'which means no rollback attack is needed, at all, when attacking these servers.'[186]

At Qualys, Ivan Ristic added that 'the impact of this problem is similar to that of POODLE, with the attack being slightly easier to execute-no need to downgrade modern clients down to SSL 3 first, TLS 1.2 will do just

---

[179] Finkle, Jim. 2014. 'New Poodle web threat not seen as menacing as Heartbleed, Shellshock.' Reuters, 15 October. As of 8 October 2015: http://in.reuters.com/article/2014/10/15/cybersecurity-encryption-poodle-idINKCN0I401X20141015
[180] Finkle, Jim. 2014.
[181] Brian, Smith. 2014. '[TLS] POODLE applicability to TLS 1.0+ (was Re: Working Group Last Call for draft-ietf-tls-downgrade-scsv-00).' Message to Bodo Muller. Internet Engineering Task Force, 17 October. As of 8 October 2015: https://www.ietf.org/mail-archive/web/tls/current/msg14058.html
[182] Brian, Smith. 2014.
[183] Brian, Smith. 2014.
[184] Langley, Adam. 2014. 'The POODLE bites again.' Imperialviolet.org, 8 December. As of 8 October 2015: https://www.imperialviolet.org/2014/12/08/poodleagain.html
[185] Langley, Adam. 2014.
[186] Pettersen, Yngve Nysaeter. 2014. 'Not out of the woods yet: There are more POODLEs.' Vivaldi Blog, 9 December. As of 8 October 2015: https://vivaldi.net/userblogs/entry/not-out-of-the-woods-yet-there-are-more-poodles

fine.'[187] Vendors such as F5, A10, Cisco, IBM (WebSphere, Domino, Tivoli), Fortinet, and Juniper were reportedly affected by the new POODLE.[188] But identifying the affected vendors was initially harder than expected. Langley noted that while his personal connections allowed him to get in touch quickly with F5, it took him almost 2 weeks to get in touch with the right person at A10.[189]

Finding solutions to the extended POODLE exploit was more difficult than getting rid of the original POODLE. Clients could not simply disable the TLS 1.x protocol as 61% of servers still only supported TLS 1.0.[190] Additionally, even servers with TLS 1.2 were vulnerable to the extended POODLE issue. Pettersen therefore offered two solutions: (1) upgrade to a customised TLS version where the issue has been patched; or (2) use an extension of the TLS protocol which was released by IETF in early 2014, which changed how encryption and integrity checking was done.[191] Vendors picked up on the first solution by releasing Security Advisories and providing patches for the devices affected. Qualys updated its free online SSL Server test in order to detect the TLS problem.[192] And TrendMicro advised its users to apply the latest Deep Security Update which helps to detect traffic from POODLE exploits.[193]

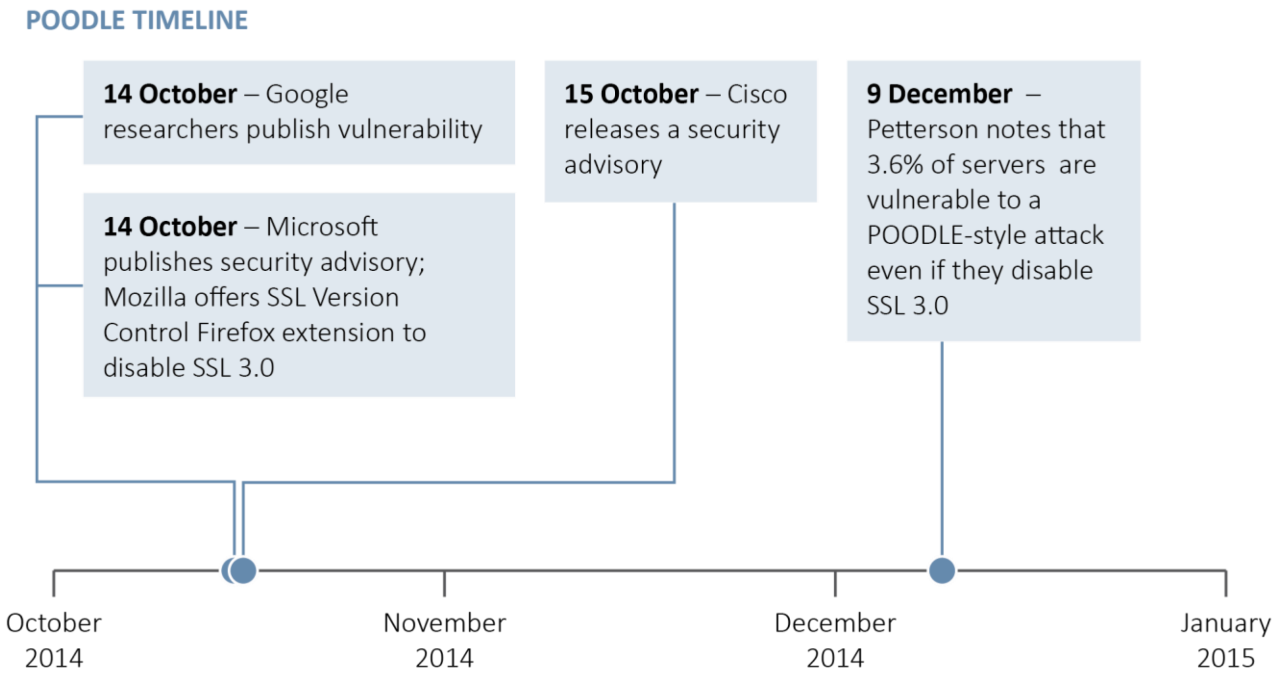


Figure 20: Timeline for the discovery and disclosure of POODLE

[187] Ristic, Ivan. 2014. 'Poodle Bites TLS.' Qualys Blog, Security Labs, 8 December. As of 8 October 2015: https://community.qualys.com/blogs/securitylabs/2014/12/08/poodle-bites-tls
[188] Langley, Adam 2014.
[189] Langley, Adam 2014.
[190] Ristic, Ivan, 2014.
[191] Pettersen, Yngve Nysaeter, 2014.
[192] Paganini, Pierluigi. 2014. 'POODLE SSL flaw is threatening also TLS Security Protocol.' Security Affairs, 10 December. As of 8 October 2015: http://securityaffairs.co/wordpress/30952/hacking/poodle-tls-flaw.html
[193] Chang, Ziv. 2014. 'POODLE More Potent, Now Affects TLS.' Trendmicro Security Intelligence Blog, 10 December. As of 8 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/poodle-more-potent-now-affects-tls/

3.5 On reflection

The four case studies have shed light on a number of challenges faced by the information security community as well as the broader public. For some of the vulnerabilities presented above, the disclosure was more a topic of discussion than it was for others. Google played a leading role in at least two of the case studies: POODLE and Heartbleed. Whereas there appeared to be no criticism of the manner of disclosure with respect to POODLE, in the context of Heartbleed, Google's role in the disclosure process was criticised by some[194]. Heartbleed, as indicated through the development of a Special Interest Group (SIG), shed extra focus on the necessity to better coordinate vulnerability disclosure across the complex landscape.

The vendor-to-vendor disclosure for Sandworm appears to have led to few problems based on available information. Microsoft and iSIGHT cooperated to track the activities of the group actively using Sandworm, which allowed them to connect information and develop a more comprehensive overview of the situation. The delayed public disclosure was based on the desire to let the disclosure coincide with Microsoft's well-known 'Patch Tuesday'.

When it comes to lessons learned, Seth Geftic, Senior Manager at RSA, concluded that POODLE has taught us that 'effective vulnerability management requires organizations to have a process in place to help very quickly identify affected systems, including the greatest risks to help prioritize remediation. Having the right tools in place to gain this visibility and quickly identify the vulnerabilities can make all the difference.'[195] Geftic further explains that while the vulnerability was in itself not dangerous, it did provide a compelling case for an intelligence-driven security strategy. Servers and clients must have a way-of-action to react immediately upon threat intelligence.[196]

Another common feature displayed by at least three of the four vulnerabilities is the lengthy period it took to discover some of these vulnerabilities. Ultimately, Shellshock happened because of a programming mistake in some computer code that went undetected for more than two decades; critical code that was being maintained by a single individual in their spare time. Heartbleed also went undetected for a number of years, as did Sandworm. The examples of Shellshock and Heartbleed pose crucial questions about the seeming over-reliance of the technology industry (and therefore businesses and consumers) on systems and products whose software has been 'built and maintained by small teams often made up of volunteers'[197] Some experts have ascribed these issues to the 'lifecycle problem' in which bugs in software are continually neglected because 'people are making mistakes whilst writing code and making further mistakes when patching the original problems.'[198] These fundamental questions require an answer from individuals beyond the information security community. Arguably, since a large number of stakeholders rely on these instruments, responsibility for their security should be borne more broadly.

Clearly, as the case studies have illustrated, there are a number of fundamental challenges associated with the disclosure of vulnerabilities. These are investigated in more detail in Chapter 4.

[194] The study team did not have an opportunity to discuss these case studies with Google to also include their perspective on the matter.
[195] Geftic, Seth, 2014.
[196] Geftic, Seth, 2014.
[197] Lee, Dave. 2014.
[198] Lee, Dave. 2014.

4. Challenges in the vulnerability disclosure landscape

4.1 Introduction

Chapter 2 has already provided an insight into the challenging landscape of vulnerability disclosure. This chapter provides more information about specific challenges identified by the interviewees of this study. The first step towards improving vulnerability disclosure is to develop an overview of the challenges that different stakeholders face. These challenges are, due to competing or conflicting interests, largely subjective. Yet, as has become evident from the interviewees' responses, certain challenges are experienced by specific stakeholders while there are several overarching issues that the broader information security community faces. Those commonalities offer room for improvement because they are shared challenges. The analysis presented in this chapter aims to highlight the challenges that have been identified by interviewees based on their experience. The interviews are therefore the primary source of data for the content of chapters 4 to 6. For reasons of continuity, the project team has tried to refrain from continually referring to the interviewees and to do so only sparingly. The analysis has also been supplemented by information gathered from the literature.

The key findings are summarised in the box below.

Key findings: what are the challenges encountered in the vulnerability disclosure landscape?

- ***Legal challenges:*** Individuals who discover a vulnerability often face legal threats when they decide to report it. These threats can have implications on not only civil and criminal law but also contract law, licensing, patent law and other types of legislation. Discoverers may find themselves in a grey area due to the methods used to discover the vulnerability and the way the vulnerability was disclosed.
- ***Lack of vendor 'maturity':*** Whereas large companies familiar within the information technology environment have robust processes in place for vulnerability reporting, other companies are new to the scene. These may be small companies, or companies which have not previously been part of the information technology landscape. This could lead to a lack of maturity on the part of these players and potentially complicate the vulnerability disclosure eco-system, as less mature companies are ill-prepared to accept vulnerability reports and act upon them in the interest of the information security community.
- ***Lack of researcher maturity:*** Just as vendors may lack experience in accepting vulnerability reports, so researchers can lack experience in reporting vulnerabilities. When such experience is absent, researchers may approach vendors in a threatening or otherwise non-conducive manner which prevents fruitful cooperation. Researchers who lack experience may also be unwilling to compromise on, for example, timelines identified by the vendor.
- **Incoming vulnerability reports are not always taken into consideration by the vendors:** Due to subjective reasons, vendors may disregard reports about a vulnerability. Vulnerabilities may be labelled as academic and theoretical and interest for a previously disregarded vulnerability report might increase after a security incident has happened.
- ***Vulnerability acquisition for national intelligence purposes:*** Unpatched vulnerabilities can be used by criminals but also potentially by national intelligence or law enforcement officials. This means that sometimes a vulnerability will remain undisclosed for such (national) security

purposes. Yet while a vulnerability remains undisclosed to the vendor, so will the development of a solution remain absent, leaving users in general vulnerable.

- ***Users do not implement patches (in a timely manner):*** Once a vulnerability is disclosed by the vendor, and a solution such as a patch to be installed via an update is available, the user must implement it. Such implementation is essential for the vulnerability to be resolved; a lack of implementation leaves users even more vulnerable since information about the vulnerability is now public. Users have a tendency, for a variety of reasons, to postpone or to negate patching altogether. This may be because of a lack of understanding or knowledge. Furthermore, it could be more important to keep the average patch application time as short as possible rather than solely focusing on the disclosure timeline.
- ***Discoverer motivation varies:*** The motivation for an individual to discover a vulnerability varies. The motivation of the discoverer can influence the decision s/he makes regarding what to do with the vulnerability. The increase in bug bounty programmes and the growing zero-day market have increasingly placed a focus on monetary reward. This may lead to over-incentivising the search for vulnerabilities and may also lead to the expectation that discoverers will always receive a monetary reward for their discovery.

4.2 Challenges

4.2.1 Legal challenges

The aim of this section is to give the reader a glimpse into some of the legal issues in the context of vulnerability disclosure (e.g. copyright law). Information provided seeks to provide general guidance to some of the legal aspects, but not as legal advice as such; certainly it doesn't constitute an exhaustive analysis of the issues at hand.

Several interviewees identified legal challenges as a primary issue of concern in the vulnerability landscape. The main source of such challenges is determined by the activities underpinning the discovery and disclosure of a vulnerability. Such activities may be formally classified as 'illegal' and as such introduce problems when the discoverer reports the vulnerability. As vendors are trying to protect their intellectual property, these kind of problems might emerge. Moussouris suggests that '*When vendors lack a process and ability to receive, investigate, remediate, and communicate about security vulnerabilities, often the first reaction is to call in the lawyers*.'[199] She describes an example where vendors made use of the Digital Millennium Copyright Act (DMCA)[200] to intimidate a security researcher. This particular case concerned security researcher Mike Davis from IOActive, who along with some colleagues discovered security vulnerabilities in electronic locks made by CyberLock. Davis and his colleagues tried to contact and disclose their findings to the company but to no avail. Instead, the researchers received a letter from the company's law firm, Jones Day, the day before they planned on publicly releasing the vulnerabilities.[201] The lawyers make reference to a provision in the DMCA which makes it illegal to circumvent digital rights technology designed to protect copyrighted work. It is perhaps useful to note in this case that the vendor sells physical devices with

[199] Moussouris, Katie. 2015. 'Vulnerability Disclosure Deja Vu: Prosecute Crime Not Research.' DarkReading, 12 May. As of 8 October 2015: http://www.darkreading.com/vulnerabilities---threats/vulnerability-disclosure-deja-vu-prosecute-crime-not-research/a/d-id/1320384
[200] United States Copyright Office. n.d.-a. 'Executive Summary Digital Millennium Copyright Act: Section 104 Report.' As of 8 October 2015: http://www.copyright.gov/reports/studies/dmca/dmca_executive.html
[201] Zetter, Kim. 2015b. 'With Lock Research, Another Battle Brews in the War Over Security Roles.' *Wired*, 6 May. As of 8 October 2015: http://www.wired.com/2015/05/lock-research-another-battle-brews-war-security-holes/

embedded software components. This vendor had not previously been confronted with a situation of vulnerability disclosure by a reporter. As such, legal challenges introduced are closely connected to the challenge of vendor maturity (see 4.2.2).

According to interviewees, when a legal team gets involved, the response can often be heavy-handed largely due to the lack of knowledge of, and sensitivity to, the technical aspects of vulnerabilities. Lawyers may immediately claim the reporter should not have looked at the code as they attempt to detach the organisation from accountability. They do not reject the existence of the vulnerabilities discovered or the insecure nature of the code, rather through legal argumentation they aim to avoid having to repair the code and patch the vulnerability. This may be due to potentially high costs associated with resolving the vulnerability.

Another concern identified is the Wassenaar Arrangement,[202] a multilateral export control regime that requires the licensing of dual-use goods and technologies including military equipment, special materials etc. In December 2014, certain types of intrusion control software were put on the dual-use goods and technologies list.[203] The wording of the control list implies that exploits are within the scope of the list. That means that anyone operating in the 41 parties[204] to the Wassenaar Arrangement would need an export licence to speak about zero-day exploits or essentially any vulnerability reporting on the internet. The fear is that the requirement of an export license will not only affect the sale of exploits, but also the existence of bug bounty programmes. Wassenaar formulates some exemptions, although not for bug bounty programmes. In contrast to the situation in the United States, the European manner of implementation may exempt bug bounty programmes. At this moment (status end of 2015), the situation is uncertain.

Overall, various stakeholders have indicated how the present and proposed legal climate in various jurisdictions does not favour improved security of the information security ecosystem.

A positive example in the EU comes from The Netherlands, where the National Cyber Security Centre (NCSC) published its guidance to promote the increase of coordinated disclosure, has made the topic an issue of public discussion. This led to a response by the Dutch Public Prosecutor's (DPP) office which sent a letter to all its departments informing them of the matter[205]. The letter indicates how the notion of ethical hacking is not a concept which can be found in criminal law in the Netherlands, nor anywhere else, most likely. The concept of ethical hacking implies that prior authorization be given by the system owner to the ethical hacker before testing the security of the systems; such an arrangement can be corroborated by a service contract directly or indirectly with the ethical hacker and the requesting vendor. While Dutch penal law does not recognise the concept of ethical hacking, the letter states that ethical motives can play a role in the determination whether an action constitutes a violation of criminal law. If a hacker finds a vulnerability and reports this vulnerability to the vendor, then this is in principle ethical hacking. However, if a hacker reports a vulnerability and there are indications that the hacker has done more, whether intentionally or unintentionally, then a criminal investigation will probably take place. 'Done more' refers for example to copying of sensitive data or personally identifying information.

---

[202] Wassenaar Arrangement. n.d. 'Introduction.' As of 8 October 2015: http://www.wassenaar.org/introduction/index.html
[203] European Commission. 2014. 'Commission updates EU control list on dual use items.' Trade, Dual Use Control, 22 October. As of 8 October 2015: http://trade.ec.europa.eu/doclib/press/index.cfm?id=1166&title=Commission-updates-EU-control-list-on-dual-use-items; see also: Official Journal of the European Union. 2014, 30 December. 'Legislation.' Vol. 57. As of 8 October 2015: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=OJ:L:2014:371:FULL&from=EN
[204] United. n.d.
[205] Openbaar Ministerie. 2013a. 'Beleid OM 'ethische hackers' in lijn met 'leidraad Responsible Disclosure.' As of 8 October 2015: https://www.om.nl/actueel/nieuwsberichten/@32028/beleid-ethische/

The main purpose of the DPP letter is to identify the grey areas of the law and assist public prosecutors in deciding whether to proceed with the prosecution of a case. In the letter there is a clear message that reporting a vulnerability does not absolve the reporter from subsequent prosecution or legal investigation. The letter has made the public prosecutor visible in the discussion, as noted by an interviewee, which is valuable in and of itself. The discoverer community responded positively towards the development because even if the letter does not provide carte blanche, it has demonstrated to the discoverer community that the public prosecutor's office is seriously thinking about the issue and how best to approach it.

In short, the letter instructs public prosecutors to take the following aspects into consideration:

Did the suspect break criminal laws in the process of finding and reporting the vulnerability?

Were the suspect's actions necessary within a democratic society, i.e. did they concern an important general interest?

Did the suspect's conduct involve proportional actions (were the means chosen in proportion to the goal to be achieved)? In other words: how did the hacker gain access to the IT system? If any disproportional actions were carried out for this purpose, this will not constitute 'ethical' hacking.

Could the discloser have taken other possible actions? In other words: was the vulnerability immediately reported to the owner of the IT system or did the discloser fail to do so in order to erase his tracks or to manipulate, copy or delete data, for example? If any tracks were erased or data manipulated, copied or deleted, this will not constitute responsible disclosure.

Figure 21: Letter from the Dutch Public Prosecutor's office on responsible disclosure[206]

The study team did not find other examples of public prosecutors providing such an explicit explanation of how they treat vulnerability disclosure cases.

4.2.2 Vendor 'maturity' varies

A central theme throughout many of the interviews was the issue of vendor maturity. Having various levels of vendor maturity presents a challenge. The difference between a mature and a less mature vendor is the ability of the organisation to cope with receiving a vulnerability report about one of its products. While discussions around vulnerability disclosure have been taking place for several years and have not fundamentally changed, new players have nevertheless appeared on the scene. A large number of vendors have not been privy to the debate as long as others, which means they are now going through the early phase of discovering how to respond to vulnerability reports. Interviewees mentioned examples of cases and companies such as car, aeroplane and refrigerator manufacturers, as well as vendors who manufacture medical devices. The latter point is crucial since, as an interviewee indicated, this goes beyond '*my credit*

[206] Openbaar Ministerie. 2013b, 18 March. Letter to: Aan alle parkethoofden, As of 8 October 2015: https://www.om.nl/publish/pages/22742/03_18_13_beleidsbrief_college_responsible_disclosure.pdf

card has been stolen'. As another interviewee stated on the topic of vendor maturity, '*While Microsoft, Cisco, Oracle, Apple and Google are familiar with that process, perhaps others are not familiar with it. They may not have a security contact point and do not understand the whole disclosure process*.' There was specific reference to the rapidly emerging Internet of Things[207] domain as an area where immature vendors are at risk in terms of vulnerability disclosure, because they are 'new to the scene' and have not reached a state where they have the infrastructure in place and the willingness to deal with vulnerability reports.

Size of the vendor also matters as a factor of influence for maturity. One of the biggest challenges is for discoverers to report a vulnerability to a small company that may not have prior experience of such a process. Discoverers who do report vulnerabilities to smaller companies often find themselves 'staring down the barrel of legal threats', as an interviewee noted (see 4.2.1).

An associated challenge with lack of vendor maturity is the ability to find the right contact person. Sometimes organisations end up speaking to someone from the marketing department rather than a more technically-inclined employee who is able to understand the vulnerability report. For organisations without security awareness, discoverers are in a difficult position in identifying the right person to speak to when reporting the vulnerability.

Maturity is required by different stakeholders to engage in vulnerability disclosure in an ethical and timely fashion. Elaborating on this point, one interviewee stated how the vendor must engage in an open and nonthreatening way to communicate, and that vendors must have a 'plan' (see 5.3.2). This also leads to a second challenge identified during the interviews: the maturity of discoverers or researchers.

4.2.3 Researcher maturity varies

The challenge of varying levels of maturity also applies to researchers or discoverers of vulnerabilities. Younger and inexperienced individuals who have not been part of the security community may not know what to do and how to report a vulnerability. They may be faced with procedural as well as ethical and legal questions. Recent graduates, as well as other younger or inexperienced reporters, may be unsure of their options and may report to vendors in a 'threatening' way. Less mature discoverers may be unreasonable in their treatment of timelines as identified by vendors. They may be more inclined to go to full disclosure without having exhausted all their options; or perhaps they may be induced into selling a vulnerability on the underground market. Responsiveness has to come from both sides.

Another interviewee echoed the challenge of having different levels of experience among discoverers. There are several main groups that find vulnerabilities and know how to handle them well. Simultaneously, there are many researchers who find one or two vulnerabilities per year and remain uncertain about how to handle them. Moreover, some researchers may also lack the time to contact the project or the vendor and properly explain the vulnerability.

4.2.4 Incoming vulnerability reports are not always taken into consideration by the vendors

Closely connected to the previous two challenges is that incoming vulnerability disclosure reports are not taken into consideration by the vendors due to subjective matters. Even with large vendors, vulnerabilities

[207] 'The Internet of Things builds out from today's internet by creating a pervasive and self-organising network of connected, identifiable and addressable physical objects, enabling application development in and across key vertical sectors through the use of embedded chips, sensors, actuators and low-cost miniaturisation.' [Schindler, Helen R., Jonathan Cave, Neil Robinson, Veronika Horvath, Petal Hackett, Salil Gunashekar, Maarten Botterman, Simon Forge and Hans Graux. 2012. *Europe's policy options for a dynamic and trustworthy development of the Internet of Things*. Santa Monica: Calif.: RAND Corporation. As of 12 October 2015: http://www.rand.org/content/dam/rand/pubs/research_reports/RR300/RR356/RAND_RR356.pdf]

which lack an immediate financial incentive may be labelled as merely academic; vendors claim that they do not believe the vulnerability will be exploited 'in the wild', which leads them to ignore the report.

This is connected to another challenge mentioned by an interviewee that concerns the cost of rectifying a vulnerability. He describes a situation where his organisation discovered a vulnerability in a car system which was not severe, i.e. would not stop the brakes from functioning or similar, but to properly fix the vulnerability required the recall of all affected cars. Such a recall is a costly endeavour, which means the manufacturer may decide, based on the severity, to tolerate the vulnerability. Yet to express such tolerance in public may be deemed unacceptable.

4.2.5 Vulnerability acquisition for national intelligence leaves users vulnerable

During the research for this report, the breach of the Hacking Team[208] came to light and served to illuminate a challenge several of the interviewees identified, albeit to varying degrees. This concerns the allegation that some governments acquire and use vulnerabilities for national intelligence and/or law enforcement purposes. According to some interviewees, this activity may pose a threat as vulnerabilities could remain unresolved, leaving users vulnerable. Further, the suspicions about law enforcement or intelligence use of vulnerabilities may impact the level of trust placed in national CSIRTs (see 2.5.2). Despite such concerns, other interviewees recognised that governments may feel it is necessary to exploit vulnerabilities to access relevant information, if it serves national intelligence or law enforcement purposes. Some may consider this acceptable, since exploitation of vulnerabilities may intrude on users' privacy less than, for example, decryption of internet traffic.

4.2.6 Users do not implement patches (in a timely manner)

Even though the challenge of getting users to actually implement the patches goes beyond the vulnerability disclosure phase, it is an essential aspect which also contains implications for how vendors handle disclosure. Despite vendor provision of information about the vulnerability and its solution, some customers still postpone the application of patches. According to an interviewee, there needs to be more trust in the vendor by the customer, especially with respect to organisations operating within critical infrastructure sectors, to apply patches as soon as possible. Any delay in patching enhances the criticality of the vulnerability since the disclosure has placed the information about the vulnerability in the public domain, allowing any perpetrator who was previously unaware of the vulnerability to attempt to exploit it (see 3.1.3.1). The importance of immediate implementation became evident through, for example, Heartbleed, where IBM security systems explained how one-day attacks can be just as dangerous as zero-day exploits.[209] Despite the existence of a patch, without its implementation users are still vulnerable. And since perpetrators are aware of the vulnerability and the potential delay of users to immediately implement patches, they could 'strike while the iron is hot.' One interviewee noted that it is more important to keep the average patch application time as short as possible rather than solely focus on the disclosure timeline. The interviewee further emphasised that the timeline is 'completed' not when vulnerability patches are made available by vendors, but when they have been applied by users. Therefore, the interviewee recommended that vendors prioritise an effective application of patches by users over a strict adherence to timelines for patch releases, in order to minimise patch 'recalls' and ensure a comprehensive vulnerability fix. Auto-update was welcomed by another interviewee precisely because users are prone to delay patching. Yet there may be other challenges associated with such a feature. Although not mentioned by interviewees, there may be other reasons for delayed patching of vulnerabilities by users, such as lack of understanding or knowledge,

[208] http://arstechnica.com/security/2015/07/massive-leak-reveals-hacking-teams-most-private-moments-in-messy-detail/
[209] Donohue, Brian. 2014.

which means that education may be a suitable means to assist users in patching vulnerabilities in a more timely manner.

4.2.7 Discoverer motivation varies

The motivation of discoverers varies, which is another challenge in the vulnerability disclosure landscape. Opinions differ on what drives researchers and others who discover vulnerabilities. According to one interviewee, there are typically three types of motivators for researchers to discover vulnerabilities. The first is a willingness to improve the security industry and the overall security ecosystem. Simultaneously, discoverers in this category will also want to develop a good track record of finding and disclosing vulnerabilities in order to enter the industry. The second category is more ego-driven. This type of researcher, according to the interviewee, is more concerned about showing off their skills to the world. The third group is seduced by the black market of vulnerability retail. Another interviewee recognises the diversity of reasons why researchers are looking for or discovering vulnerabilities. According to him, the community needs to be able to cater to all of the reasons that may drive researchers. He, however, believes that only a few are focused on monetary rewards. This is a point of contention since money does seem to play a key role. As another interviewee commented, 'There is a small concern, and I don't know if this will bear out, but if it is common practice to always pay for a vulnerability, could that turn the incentives around and people would be less likely to disclose because they are not getting enough money or because they cannot find a buyer?' There is at least anecdotal evidence for this. As described by Perlroth, '*simply crediting hackers or sending them swag no longer cuts it*.'[210] Ramses Martinez, Yahoo's director of security, said he started Yahoo's bounty programme in 2013 after two hackers criticised Yahoo for sending them T-shirts in exchange for four bugs that could have brought them thousands of dollars on the black market. Martinez says he now considers bug bounties a "no-brainer".'[211]

There are other concerns about paying for vulnerabilities. Through the introduction of bug bounty programmes and vulnerability reward programmes, as well as the zero-day market, there are financial incentives to discover and sell vulnerabilities. This may lead to an oversupply of vulnerabilities as the market develops further. Therefore, caution is advised to avoid over-incentivising the search for vulnerabilities because it may lead to a high number of 'minor' vulnerabilities which once discovered need to be addressed. The overwhelming number may then divert scarce resources from more critical vulnerabilities which can cause greater harm.

[210] Perlroth, Nicole. 2015.
[211] Perlroth, Nicole. 2015.

5. Good practices for stakeholders active in vulnerability disclosure

5.1 Introduction

To establish a bridge between challenges and recommendations for improvement, this chapter explores a selection of good practices identified by interviewees. These good practices are often closely associated with the challenges and the recommendations for improvement, which means there is some overlap. This chapter will allow different stakeholders to better understand what practices they can implement with the intent of improving the overall vulnerability ecosystem. There is a primary focus on vendors and discoverers.

A list of the good practices identified have been summarised in the box below:

Good practices for stakeholders active in the vulnerability disclosure landscape

- ***Use existing documents:*** Previous efforts have been made to gather good practices in the area of vulnerability disclosure as well as to describe how to set up a vulnerability disclosure policy. To prevent reinvention of the wheel, these documents should be used by stakeholders and should be leveraged more by new initiatives. The ISO standards are a prominent example, although their lack of free availability may hamper their reach.
- ***Communication***

  Vendors should be reachable/have a point of contact: To prevent reporters from having to spend valuable time and resources looking for the appropriate contact, vendors should have a clear point of contact to deal with vulnerability reports, and this contact should be reachable.

  Have a policy in place: Vendors should have a policy in place which addresses vulnerability disclosure and describes how they respond to vulnerability reports. This policy will also indicate to reporters what information they need to provide as well as give an insight into the process of the disclosure.

  Communication with different stakeholders: Communication with stakeholders requires mutual respect, patience and transparency. Continual communication is essential to acknowledge receipt of the vulnerability report as well as provide an indication of the next steps.
- ***Information dissemination:*** Information about the vulnerability as well as its solution, if available, should be disseminated to inform users of the developments and to provide them with an opportunity to protect themselves. How much information needs to be disseminated is a topic of discussion among stakeholders.
- ***Timelines:*** Timelines vary but a timeline should be mutually agreed upon (on a vulnerability-by-vulnerability basis) to ensure that a vulnerability will be sufficiently addressed by the vendor in a timely manner.
- ***Flexibility in reporting and disclosing:*** No one size fits all in the area of vulnerability disclosure, so flexibility is necessary to tailor the vulnerability report as well as the response to the specifics of the vulnerability.

5.2 Use existing documents

When asked about good practices for vulnerability disclosure, several interviewees identified existing documents which contain good practices. One example is the Organization of Internet Safety (OIS) document.[212] According to an interviewee this provides a 'pretty good framework' for how responsible disclosure ought to be carried out. Simultaneously, though the interviewee recognised the need for refreshment of the document, he emphasised how the community should not 'reinvent the wheel' but rather use what already exists and share it with stakeholders in a more active manner. Better-known examples of good practice frameworks for vulnerability disclosure are two International Standards Organisation (ISO) standards. These are ISO 29147[213] and ISO 30111,[214] which one interviewee recognised as 'good reference documents.' The intention of both standards is to inform vendors who are in the process of creating a vulnerability-handling and disclosure scheme. Larger vendors, governments and enterprises helped to put the standards together. Some interviewees identified the ISO standards and appeared to credit them with some value, whereas another interviewee stated how the ISO standard has had 'very little' impact. Although the interviewee recognised how these good practices, shared by the larger vendors, are successful, their impact is limited. The value of the ISO standard is in bringing these practices to smaller vendors, but this comes at a cost. ISO standards are not free and access requires payment of a fee. The interviewee identified this as a big problem, 'because in security information technology pretty much all the standards are free, open, and available. It is thus to ISO's detriment that its standards are not freely available. If they were they would be much more usable for smaller organisations.' Moreover, the interviewee stated how he did not believe there was a lot of guidance on good practices for vulnerability disclosure. There are other relevant documents such as the Internet Engineering Task Force (IETF) Responsible Vulnerability Disclosure Process;[215] vulnerability disclosure processes of experienced organisations may also help to illuminate some of the aspects for less mature organisations.[216]

This study aims to therefore enhance the availability of such guidance on good practices and to stimulate improvement of the vulnerability disclosure situation through the dissemination of such practices as well as the identification of challenges and associated recommendations for improvement.

---

[212] Organization for Internet Safety. 2004.
[213] ISO/IEC 29147:2014 provides guidelines for the 'disclosure of potential vulnerabilities in products and online services. It details the methods a vendor should use to address issues related to vulnerability disclosure.' [ISO. 2014. 'Information technology -- Security techniques -- Vulnerability disclosure.' As of 14 August: 2015: http://www.iso.org/iso/catalogue_detail.htm?csnumber=45170]
[214] ISO/IEC 30111:2013 provides guidelines for 'how to process and resolve potential vulnerability information in a product or online service.' [ISO. 2013. 'Information technology -- Security techniques -- Vulnerability handling processes.' As of 14 August 2015: http://www.iso.org/iso/catalogue_detail.htm?csnumber=53231]
[215] Christey, Steve & Wysopal, Chris. 2002.
[216] Relevant examples include: Microsoft: http://go.microsoft.com/?linkid=9770197
Google: https://www.google.com/about/appsecurity/; http://googleonlinesecurity.blogspot.be/2015/02/feedback-and-data-driven-updates-to.html
Apple: https://www.apple.com/support/security/
Facebook: https://www.facebook.com/whitehat
Oracle: http://www.oracle.com/us/support/assurance/vulnerability-remediation/disclosure/index.html
Lenovo: https://support.lenovo.com/us/en/documents/ht103338
Rapid7: https://www.rapid7.com/disclosure.jsp
Zero Day Initiative: http://www.zerodayinitiative.com/advisories/disclosure_policy/
Bugcrowd: https://bugcrowd.com/resources/responsible-disclosure-program
HackerOne: https://hackerone.com/disclosure-guidelines
CERT/CC: http://www.cert.org/vulnerability-analysis/vul-disclosure.cfm; https://forms.cert.org/VulReport/
NIST: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf

5.3 Continuous communication is essential

5.3.1 Vendors should be reachable/have point of contact

At the bare minimum, vendors should be reachable and have a primary point of contact. Organisations responsible for engagement with reporters (or potential reporters) of vulnerabilities spend a lot of time and energy trying to find the right contact. This has led to at least one organisation, according to its own testimony, sending recorded physical mail to locations so they can obtain a return receipt, which makes it more difficult for vendors to ignore them.

5.3.2 Have a specific policy in place to deal with the disclosure process

There is a preference to go beyond merely a primary point of contact and for vendors to have a specific policy in place that details how they handle vulnerabilities which discoverers report to them. It was noted that such a policy should also include another good practice mentioned, which is communication with the discoverer (see 5.3.3). A specific policy must at least address the following aspects:

- Point of contact
- Information required from the reporter
- Possible responses from the vendor
- Timeline of the process.

Closely related is the necessity – as an organisation – to have good arrangements with suppliers. This is because sometimes a vulnerability can surface in a third-party system which is outside the user's direct control, and to act quickly arrangements with such suppliers and service providers must be in place to determine how to handle the situation.

5.3.3 Communication with different stakeholders

Communication with the different stakeholders is key in terms of good practice. With respect to discoverers, communication with the discoverer is crucial to ensure the process does not lead to unexpected outcomes. Vendors need to be very clear about their expectations with regard to discoverers. Expectations should be made explicit by companies, preferably through a statement on their website. Once a vulnerability has been reported to a vendor, the company should respond as soon as possible and then keep the discoverer in the loop as far as is practicable for the vendor or coordinator, and desirable for the discoverer. The OIS states that within a maximum of seven calendar days of receiving a finder's report, the vendor should acknowledge its receipt.[217]

Even if a fix is not (yet) available, it is important to communicate the status to the discoverer and to acknowledge that the organisation is working on the issue. Just as communication with the discoverer is a central element, so is communication with the vendor, as well as with other companies.

5.4 Information dissemination must occur, but opinions differ on how much

With respect to users of products, publishing information about vulnerabilities is crucial. Many vendors worry about publishing vulnerabilities discovered in their products, out of fear that such publication reflects poorly on them; as if they have made a mistake and criminals or others with malicious intent will take advantage of it. An interviewee describes how he and his colleagues respond by stating: '*First you are not alone, no one writes perfect software. All software has security vulnerabilities. The maturity marker is that you actually acknowledge them and fix them quickly. And provide your customers with a fix. That is what you want to be measured on and not the fact that you had 10 bugs or zero-bugs or whatever*.' He goes on to

[217] Organization for Internet Safety. 2004.

state that even if a vendor decides not to take any action or disagrees on the existence of a vulnerability, it is important to be transparent.

The decision to publish information is driven by the desire to reduce the impact of the vulnerability on society, whether business or consumers. Yet with users not implementing patches in a timely manner (see 4.2.6), the dilemma becomes how much information to share since it is difficult – as noted by an interviewee – '*to share enough to enlighten the public and not to stimulate attackers to use it or let them figure out where they would get that further intelligence information*.' Another interviewee emphasised the need to provide only the information that users actually need to determine whether the vulnerability affects them; and to understand the impact if they decide not to apply the vulnerability patch. Technical details, for example, are not provided to users because it is not necessary, according to the interviewee. This is, however, a source of disagreement. Determining the appropriate amount of detail to disclose was also described by another interviewee as a challenge. According to him, however, the decision how much information to disclose is up to the researcher rather than the vendor, at least after the vulnerability has been addressed. Another interviewee explicitly stated how he would like all open source vendors to provide the technical details about the vulnerabilities they fix. He describes further how, *'Some vendors operating on the limits of open and closed source definitions have secretive practices that effectively harm the community's and other vendor's abilities to assess and report their own versions of those software packages. We are left with "open" projects that are effectively secretive dumps of code with no explanations*.'

Information dissemination is therefore considered good practice, but how much, to whom and in what form, remains an area of discussion. Zetter describes a situation between FireEye and ERNW – both information security firms. ERNW described in a blogpost in September 2015, according to Zetter, how FireEye issued a court injunction against ERNW. Even though FireEye agreed that ERNW could disclose its discovery of three vulnerabilities in FireEye products, the dispute concerns the amount of information ERNW can publicise about the vulnerabilities.[218] Perhaps the question of how much information to disseminate ought to be carefully assessed on case-by-case basis by each party involved in the process. Arguably, information output needs to be improved because, as an interviewee indicated, companies are drowning in information. Information needs to be more tailored. As one interviewee described it, 'Let's be clear on what it is. Whether it's important or not and what you need to do. Otherwise we are spinning wheels for nobody's benefit. Just that difference between do I have to update today, is it real, is it relevant, is it theoretical, do we think it can get real…we need to distil it down. The world is drowning in information overload.'

5.5 Timelines lead to results

There appears to be a consensus that timelines are a necessary part of vulnerability disclosure and can therefore be identified as a good practice. Without timelines, certain vendors may not address the vulnerability at all. Providing vendors with short embargos for the development of a solution keeps them 'on top of their game', since they cannot 'sit' on a vulnerability for months without a response. While a timeline is not a guarantee that a vendor will address the vulnerability, it does often lead to results, i.e. an actual response from the vendor and subsequent action to address the vulnerability.

It is, however, also important to note that when reporters disclose unfixed vulnerability information upon the expiration of a timeline, users and customers of corresponding vendors are exposed to increased risk of attack. Therefore, there needs to be agreement among the vendor community about reasonable timelines to address a particular vulnerability, and continued collaboration with the reporter in addressing the vulnerability. Timelines must take into account the different nature of distinct vulnerabilities as well as the

---

[218] Zetter, Kim. 2015c. 'A Bizarre Twist in the Debate over Vulnerability Disclosures.' *Wired*, 11 September. As of 8 October 2015: http://www.wired.com/2015/09/fireeye-enrw-injunction-bizarre-twist-in-the-debate-over-vulnerability-disclosures/

products in which they are discovered. They must involve regular communication with the reporter to ensure a degree of flexibility, should unexpected complications arise. As an interviewee said, 'one timeline for all type of vendors is unrealistic' and timelines should be agreed between vendors and reporters on a case-by-case basis. The OIS also testifies to this in its Guidelines for Security Vulnerability Reporting and Response, when it writes: '*There is no single universally appropriate timeframe for investigating and remedying security vulnerabilities*.'[219] Timelines help facilitate a timely response. Large vendors are reasonably good at acknowledging vulnerabilities, working to generate solutions, and disclosing the information (see 4.2.2). Wider adoption of this approach is needed by other organisations and industries to harmonise and streamline the process.

Project Zero[220]
Even though this report aims to refrain from endorsing any particular initiative introduced by vendors, some interviewees specifically identified Google's Project Zero as a potential example of 'good practice.' Project Zero is a security research team at Google consisting of individuals who are specialised in vulnerability research and software exploitation. The team's mission is to 'make 0day hard.' This is motivated by the observation that (1) software exploits are increasingly traded in a private market that is not accessible to software vendors and open-source projects, and that (2) software exploits traded in this manner result in user harm. Project Zero has discovered and managed disclosure of over 250 security vulnerabilities in a wide variety of products since the team's formation in July 2014. The main good practice identified by interviewees about Google's Project Zero is the clarity, especially with respect to the Project's timeline. Project Zero uses a 90-day disclosure deadline on all of its vulnerability reports. If 90 days elapse without a broadly available patch, then the bug report will automatically become visible to the public. There are exceptions however. In February 2015, Project Zero updated its policy to include a grace period of 14 days.[221] As the team writes in a blogpost, 'If a 90-day deadline will expire but a vendor lets us know before the deadline that a patch is scheduled for release on a specific day within 14 days following the deadline, the public disclosure will be delayed until the availability of the patch. Public disclosure of an unpatched issue now only occurs if a deadline will be significantly missed (2 weeks+).'[222] The Project Zero team believes that on balance this is the optimal approach for user security. The timeline provides vendors with a 'fair' and 'reasonable' length of time to engage in the vulnerability management process but simultaneously acknowledging the urgency introduced by private bug collisions.

5.6 Flexibility of reporting and disclosing

The timeline discussion has already briefly indicated how flexibility in terms of the way a vulnerability is reported and subsequently treated is key, since no one size fits all vulnerabilities and their disclosures. Interestingly, however, flexibility – according to some – should be on the side of the discoverer rather than the vendor. From this perspective, the discoverer is at liberty to determine what flexibility is required for the vulnerability. Yet flexibility, just as responsiveness, ought to be a two-way street to ensure there is common ground for achievement of the ultimate outcome. The discoverer may, for example, be insufficiently aware of the effort needed to develop a patch, and the implications of its release. Flexibility is also required, for example, with respect to patching for critical infrastructure sectors, which is more complicated and therefore requires more time for vendors to develop a patch. Flexibility is also necessary for other

[219] Organization for Internet Safety. 2004.
[220] For more info: https://cansecwest.com/slides/2015/Project%20Zero%20-%20making%200day%20hard%20-%20Ben%20Hawkes.pdf
[221] Evans, Chris, Ben Hawkes, Heather Adkins, Matt Moore, Michael Zalewski & Gerhard Eschelbeck. 2015. 'Feedback and data-driven updates to Google's disclosure policy.' Project Zero, 13 February. http://googleprojectzero.blogspot.com/2015/02/feedback-and-data-driven-updates-to.html
[222] Evans et al., 2015.

stakeholders, such as users or third parties involved in the development and subsequent dispatching of the patch.

6. Recommendations for improvement

6.1 Introduction

The exploitation of vulnerabilities discovered in systems will continue to pose security risks, with potentially damaging economic and societal impacts. Therefore it becomes all the more important that the various stakeholders involved in this complex environment attempt to address – together – the various challenges that are encountered in the vulnerability disclosure landscape. From the perspective of the security and trust of the end-users of systems, there is consensus that vulnerabilities must be disclosed in a way that minimises damage. Although movement towards more and better coordinated vulnerability disclosure has been happening to some extent for several years now, the landscape is still 'fragmented' in many ways, and there are pertinent questions that remain unanswered. The synthesis of the various sources of evidence used in our analysis shows that there are several areas for consideration. This chapter outlines these recommendations for improving the status quo in the vulnerability disclosure landscape. It is important to highlight that our intention is not to prescribe detailed and specific actions, but rather to raise the key issues which require careful consideration by all stakeholders involved in the vulnerability disclosure community. For every recommendation, the study team has also identified the potential role for ENISA, which is a determination based on input from interviewees as well as analysis from the team.

The core recommendations for improvement have been summarised in the box below.

Core recommendations for improvement from the analyses

- ***The community must facilitate the improvement of vendor maturity:*** To make progress, vendor maturity must be improved to ensure that all vendors are able to receive vulnerability reports and respond to them in a manner which is accepted by the community and which will introduce the smallest risks with respect to the security of users. In this context the term community refers to different relevant stakeholders like EU Member States, vendors, security researchers, national CSIRTs and ENISA. To improve vendor maturity, the community must stimulate less mature vendors to introduce a policy and set up an infrastructure which allows them to accept vulnerability reports.
- ***Internationalisation through policy learning:*** The global nature of the internet requires a more transnational approach to the topic of vulnerability disclosure, where successful cases in certain countries or regions can be used for policy learning purposes in other areas of the world. Simultaneously, stakeholder gatherings at the transnational level can use their international access to further enhance such policy learning, and so allow the spread of good practices in vulnerability disclosure.
- ***Introduction of a neutral third party or enhancement of existing coordination centres:*** The different interests held by stakeholders – especially reporters and vendors – as well as the growing complexity of the landscape, both in terms of stakeholders and products, advocates the introduction of a neutral third party to coordinate vulnerability disclosure. An alternative is to enhance existing coordination centres, to ensure that power discrepancies as well as potential conflicts of interest will not compromise the overarching goal of improved information security.
- ***European policy makers and Member States should improve the legal landscape:*** The current legal framework poses challenges for stakeholders involved in the disclosure process, as varying requirements across jurisdictions need to be aligned, in order to facilitate information security and

stimulate market growth. In this direction the current legal framework needs to be reviewed for requirements that pose obstacles rather than facilitating the processing of vulnerabilities.

- ***Vendors should facilitate trust building, transparency and openness***: From a vendor's perspective, the stigma associated with acknowledging that one of its products contains a vulnerability could potentially lead to an unwillingness to recognise the existence of vulnerabilities. Society should therefore move towards a state where the existence of vulnerabilities is acknowledged and accepted, to facilitate more openness as a precursor to improved information security.
- ***ENISA could facilitate and advise to improve the vulnerability disclosure landscape:*** ENISA could play a facilitating and advisory role in the area of vulnerability disclosure through information dissemination, providing recommendations, striving for harmonisation, collaborating with the security researcher community and demonstrating leadership. From a policy perspective, ENISA could advise the European Commission about the necessity for transparency from vendors and the negative impact of copyright law in the EU.

6.2 The community must facilitate the improvement of vendor maturity

As noted in Chapter 4, varying levels of vendor maturity is a problem for the overall vulnerability disclosure landscape. Therefore, one of the recommendations is to improve vendor maturity. Such maturity can be enhanced practically through implementing some of the good practices identified in Chapter 5. The main recommendation for vendors is to have a policy in place. This will enhance the level of vendor maturity and improve communication between stakeholders when vulnerabilities are discovered and reported. A disclosure policy also improves transparency, enhances the quality of the products sold by vendors, and stimulates the researchers as well as the broader information security community. Due to the necessity of having to act quickly, it is essential to have a policy in place, otherwise coordinated vulnerability disclosure will probably not work. Witnessing more vendors introducing policies and starting to think about how they receive vulnerability reports is 'the biggest positive change that we have seen over the last couple of years', according to an interviewee. Overall, vendors from different sectors could be encouraged & supported by the Member States, national CSIRTs, and ENISA to implement a disclosure policy. These include organisations within sectors that are increasingly more reliant on digital technology such as car manufacturers, airlines and maritime, plus organisations generally included in the Internet of Things and Industrial Control Systems domains. Besides private sector organisations, governments also need to have a robust vulnerability disclosure process in place. Such a policy ought to combine both defensive and offensive elements, while erring on the side of disclosure. More vendors ought to start practising vulnerability disclosure in any shape or form, but preferably in a streamlined and harmonised manner through a published and clear policy.

ENISA can potentially assist in improving vendor maturity through Information dissemination about the vulnerability disclosure process. This includes the development of a vulnerability disclosure policy and the implementation of such a policy in the organisation.

ENISA could provide leadership, particularly around exploring the issues and perhaps bringing together information from all stakeholder groups, publishing that information, and providing good practices. Because of its impartiality and its pan-European reach, bringing common understanding is a step in the right direction.

6.3 Internationalisation through policy learning

Due to the global nature of the internet, coordinated vulnerability disclosure needs to be adopted at an international level. A mechanism needs to implemented to engage in policy learning from countries where a particular model of coordinated vulnerability disclosure, such as in the Netherlands, has led to positive results.[223] The Global Forum on Cyber Expertise, which began after the Global Conference on Cyber Space (GCCS) in April 2015 in The Hague, the Netherlands, was suggested as a viable venue in which to do that.[224] ENISA was also recommended as an option although its reach may be limited to the EU context. Simultaneously, the Security Interest Group (SIG) started as part of Forum of Incident Response and Security Teams (FIRST) could also present a viable alternative, since the SIG gathers various stakeholders to agree a standard of principles on the topic of coordinated vulnerability disclosure.

ENISA could use its position and network to strive for harmonisation of vulnerability disclosure processes.

FIRST introduces Special Interest Group (SIG) to come to coordination principles
Largely as a result of Heartbleed, the challenge of coordination prompted the introduction of a Special Interest Group (SIG) as part of FIRST. The Industry Consortium for Advancement of Security on the Internet (ICASI)[225] initiated the SIG because of its belief that all stakeholder communities needed to be involved in the discussion. The main focus is on developing a consensus on how to improve coordination. ICASI itself was too limited in its membership. As a result, the initiators turned to FIRST with the request of sponsoring a SIG on the topic of vulnerability coordination. The SIG has two aims: first to "'drive the industry towards more organised and repeatable approaches to vulnerability coordination'; and secondly to 'routinely share best practices and exchange protected information around vulnerabilities and mitigations.'[226] With sponsorship from FIRST, the stakeholder communities involved in the SIG are growing: from vendors, researchers, CSIRTs, open source communities, to bug bounty programmes. Membership is open to anybody. The SIG currently faces a relative underrepresentation of the open source community. Since stakeholder engagement is deemed a priority, the SIG seeks to address the problem of disproportionate representation by targeted outreach. At the FIRST conference in Berlin in 2015, the SIG ratified two thrusts of effort. The first effort focuses on improving coordination. This includes elements such as developing an understanding of whom to go to and who would be a stakeholder that may be impacted. The SIG also aims to research whether there is a directory, or some type of resource, that a researcher or company could access if they find a vulnerability. To accomplish this first effort, the SIG wants to explore the possibility of a coordination directory. This requires the group to gather information on existing directories and whether FIRST can connect to such an existing directory, or whether it is preferable to establish a new one. Running parallel to that is the second effort around establishing a set of coordination principles. Once agreed, the principles would be published with use cases, approximately 3-5, that explore those principles. The use cases are both representative studies from real-life and hypotheticals created to tease out the complexities of vulnerability disclosure. The dialogue around the use cases will provide a 'non-threatening, neutral forum' while highlighting potential dissension from various communities on how best to coordinate. The SIG aims to present its

[223] National Cyber Security Centre. 2015. 'Introducing Responsible Disclosure: Experiences in the Netherlands: A Best Practice Guide.' As of 8 October 2015: https://www.gccs2015.com/sites/default/files/documents/BestPracticeRD-20150409_0.pdf
[224] Global Conference on Cyberspace 2015. n.d. 'Global Forum On Cyber Expertise (GFCE).' As of 8 October 2015: https://www.gccs2015.com/gfce
[225] ICASI has previously published: Schiffman Mike. 2011. 'The Common Vulnerability Reporting Framework: An Internet Consortium for Advancement of Security on the Internet (ICASI) Whitepaper.' As of 8 October 2015: http://www.icasi.org/wp-content/uploads/2015/06/cvrf-whitepaper.pdf
[226] ICASI. 2015. 'Current Activities.' As of 17 August 2015: http://www.icasi.org/current-activities/

findings during the FIRST conference in 2016 in Seoul. The form of the final 'product' depends on the process.

6.4 Introduction of a neutral third party or enhancement of existing coordination centres

The project team's analysis sees merit in having a neutral third party to whom vulnerabilities can be reported, especially since certain entities may hold vested interests which can potentially conflict with the overarching interest of a secure ecosystem. Some interviewees voiced suspicions of reporting vulnerabilities to certain National CSIRTs, since such CSIRTs are governmental entities which, due to the interest of national intelligence in vulnerabilities, may be influenced by other government departments. Yet, 'given the fact that there is usually a very big power discrepancy between the vendor and the security researcher, I actually think it makes sense to bring in another party with experience that can help with those discussions.' This can be CSIRTs or another party, although certain existing CSIRTs maintain significant experience as coordinators of vulnerability disclosures.

There is a consensus among interviewees that there are three CSIRTs which have extensive experience with coordinating vulnerability disclosure. These are CERT-CC in the United States, JP-CERT in Japan and CERT-FI in Finland. The work carried out by these coordination centers is widely recognized and it is recommended that since they already have the 'know-how', they should continue to lead these activities. Simultaneously, the involvement of other national-level CSIRTs or third-party vulnerability coordinators would benefit the ecosystem, since they are able to reach a wider audience. Other benefits include limiting the legal exposure of the discoverer, allowing the discoverer to (potentially) remain anonymous, and the increased likelihood of vendors to respond in a timely manner when coordination centers are involved.

The dilemma therefore is whether the introduction of a neutral third party is more desirable than enhancing existing efforts. As another interviewee commented, 'Introducing a new institution that would do global co-ordination might be an option but the composition, placement and the process would have to be very carefully considered.'

ENISA could provide recommendations on how to communicate vulnerability disclosures, particularly in terms of which companies and vendors must be informed.

6.5 European policy makers and Member States should improve the legal landscape

The legal framework presents many challenges. The overall impression is that existing law, especially the way legal representatives of vendors interpret and subsequently use such law, adversely affects security. One of the concerns refers to the legal liability of the vulnerability discoverer. As previously described, the potential legal threat in the area of criminal law can have a chilling effect. In the Dutch example, the public prosecutor's office has tried to illuminate its decision-making process in this area through the publication of a letter which indicates what questions public prosecutors ask before deciding whether charges will be filed against a discoverer. Other governments can follow a similar example since it helps the research community better understand the decision-making process and allows them to critically evaluate their own actions. Even though the ethical hacker acts on behalf of the system owner he/she cannot escape the legal grey area altogether, especially since there are no legal provisions – as far as is known – which specifically identify exceptions for such hacking. As a result, clarity can only be achieved on a case-by-case basis. The Dutch public prosecutor's office used case law to arrive at its explanation of how it approaches this type of legal dilemmas. Case law has set the framework on how to consider potential challenges in the area of vulnerability disclosure. By using case law, vulnerability disclosure cases can become more predictable for discoverers and other stakeholders involved. This will hopefully lead to less uncertainty about potential legal consequences.

In the United States, the main challenge in the area of criminal law comes through the Computer Fraud and Abuse Act (CFAA), which disincentives the reporting of security vulnerabilities.[227] . There is criticism about the vague wording of the CFAA which allows for considerable prosecutorial discretion, leaving researchers in uncertainty when they discover a vulnerability.[228] There is currently discussion about introducing exemptions to the CFAA for security researchers.[229] A similar solution is being discussed for the DMCA.[230]

Within the EU, member states have different interpretations of how to approach reported vulnerabilities from a criminal law perspective. As indicated in Chapter 6, the public prosecutor's office in the Netherlands has indicated how the office approaches the issue of vulnerability reporting from a criminal law perspective. Such an approach provides an example that could be used by other EU member states to establish a level of clarity and transparency, and thus develop a degree of confidence among researchers in determining how their actions fit within the existing legal framework. Evidence for other Member States being as explicit on this topic, at least from a public prosecutorial perspective was not identified by the study team during its research. The overview provided by Biancuzzi does however confirm that many Member States do not have specific legislation on vulnerability disclosures which means case law and interpretation of the current legal framework are likely to guide any future steps.[231] This can lead to uncertainty unless – as in the Netherlands – public prosecutors provide insight into how they approach cases of vulnerability disclosure reporting. The common risks that a vulnerability researcher is facing while seeking for vulnerabilities without prior authorization could be associated with the abuse of intellectual property rights, breach of licensing as well as other types of legal liabilities. A similar recommendation was set forth by ENISA in 2013 with respect to the Directive on attacks against information systems, where the authors write with respect to illegal access: 'To reduce legal uncertainty, it could be advisable for countries to publish guidance on the interpretation and application of the unlawful access provisions, and particularly on the element of intent (i.e. the unlawfulness – without right) in cases where no security measures were breached, if this is permitted under national law. This can be done in the form of prosecution guidelines in countries that permit this, and/or in the form of jurisprudence overviews to show how courts apply the law in reality. Collection and dissemination of such guidance at the EU level could also help to ensure homogeneous application of the law across the European territory.'[232]

---

[227] Kirsch, Cassandra. n.d. 'The Grey Hat Hacker: Reconciling Cyberspace Reality and the Law.' As of 8 October 2015: http://www.nku.edu/content/dam/chaselaw/docs/centersandinstitute/Law---Informatics/Symposium-CLE/Kirsch_Working%20Paper_The%20Grey%20Hat%20Hacker.pdf

[228] Ellis, Jen. 2015. 'How Do We De-Criminalize Security Research? AKA What's Next for the CFAA?.' Rapid 7 Community, 26 January. As of 8 October 2015: https://community.rapid7.com/community/infosec/blog/2015/01/26/how-do-we-de-criminalize-security-research-aka-what-s-next-for-the-cfaa

[229] United States Copyright Office. n.d.-b. 'Exemption to Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies Under 17 U.S.C. § 1201.' As of 8 October 2015: http://copyright.gov/1201/2015/post-hearing/answers/Class_25_Hearing_Response_CDT-OTI_Docket_No_2014-07_2015.pdf

[230] Stallman, Erik. 2015. 'The Current DMCA Exemption Process is a Computer Security Vulnerability.' Centre for Democracy and Technology, 21 January. As of 8 October 2015: https://cdt.org/blog/the-current-dmca-exemption-process-is-a-computer-security-vulnerability/

[231] Biancuzzi, Federico. 2008. 'The Laws of Full Disclosure.' Security Focus, 26 February. As of 8 October 2015: http://www.securityfocus.com/columnists/466/2

[232] De Muynck, Jo, Hans Graux & Neil Robinson. 2013. 'The Directive on attacks against information systems: A Good Practice Collection for CERTs on the Directive on attacks against information systems.' ENISA. As of 8 October 2015: https://www.enisa.europa.eu/activities/cert/support/fight-against-cybercrime/the-directive-on-attacks-against-information-systems/at_download/fullReport

In the context of the civil law, some concerns refer to the copyright legislation, which certain legal representatives of vendors use to threaten discoverers who report vulnerabilities. A critical evaluation of copyright law must be carried out at the EU level to determine whether amendments have to be made to ensure the legal climate does not unnecessarily obstruct security research. Alternatively to amendments, if deemed unnecessary, the current legal framework must be made more accessible and understandable for a non-legal audience, such as discoverers, to be able to appropriately defend themselves as well as their interests when receiving legal threats. The EFF, for example, provides an overview of all the applicable areas of law in the United States with respect to reverse engineering.[233] In May 2012, the European Court of Justice (ECJ) ruled how 'there is no copyright infringement' when a software company without access to a program's source code 'studied, observed and tested that program in order to reproduce its functionality in a second program.'[234] Even though in the aforementioned example there was no breach of the copyright legislation, there could be potential breaches of other legal requirements imposed by means for e.g. contract terms, confidentiality, privacy, property etc. The ECJ furthermore states, 'this means that the acts of loading and running necessary for the use of a copy of a program which has been lawfully acquired, and the act of correction of its errors, may not be prohibited by contract.'[235] Correction of its errors may mean that vulnerability-related research is exempt from copyright provisions, or rather does not infringe on copyright. Yet, more clarity is needed to definitely ascertain this and provide legal certainty to discoverers. The decision was upheld in the United Kingdom by the Court of Appeal in 2014.[236] Law is inherently about the appropriate distribution of rights and responsibilities; as such while vendors have rights, they also have responsibilities, just as researchers do.

Another challenge refers to the zero-day market, especially in connection to law enforcement and intelligence. Vulnerabilities are in some cases exploited for security purposes, which leads to an inherent conflict of interest since their prolonged existence also results in a state of insecurity. A public discussion on the issue on how society believes vulnerability disclosure ought to take place could be a way forward in this hard to resolve issue (see 6.6).

ENISA could advise the European Commission and the EU Member States on the impact that different legal frameworks have on vulnerability research and reporting, and assist in developing a more conducive legal framework.

6.6 Vendors should facilitate trust building, transparency and openness

There is a general need to engage in trust building, transparency and openness of vulnerability disclosure. This requires a societal acceptance that vulnerabilities are an inherent aspect of product development. From a vendor's perspective, the stigma associated with acknowledging that one of its products contains a vulnerability could lead to an unwillingness to recognise the existence of vulnerabilities. As one interviewee described it, 'A lot of vendors we have talked to are understandably concerned in the beginning about publishing a problem in their software. They fear that it looks bad, that they made a mistake, and that the bad guys will find out and hurt their customers with it.' Therefore, any discussion of vulnerability disclosure

[233] Electronic Frontier Foundation. n.d. 'Coders' Rights Project Reverse Engineering FAQ.' As of 8 October 2015: https://www.eff.org/issues/coders/reverse-engineering-faq#faq4
[234] InfoCuria – Case –Law of the Court of Justice. 2012. 'JUDGMENT OF THE COURT (Grand Chamber).' As of 8 October 2015: http://curia.europa.eu/juris/document/document.jsf?text=&docid=122362&pageIndex=0&doclang=en&mode=req&dir=&occ=first&part=1&cid=564907
[235] InfoCuria – Case –Law of the Court of Justice. 2012.
[236] Wood, Ian. 2014. 'UK: Court Of Appeal Upholds High Court Decision In SAS Institute Inc V World Programming Ltd Software Infringement Case.' Mondaq, 4 April. As of 8 October 2015: http://www.mondaq.com/x/304618/Copyright/Court+Of+Appeal+Upholds+High+Court+Decision+In+SAS+Institute+Inc+V+World+Programming+Ltd+Software+Infringement+Case

must stress the idea that vulnerabilities are an inherent part of software and hardware products. There should be no need to hide the fact that vulnerabilities exist.

ENISA could facilitate outreach and collaboration with the security researcher community.

ENISA could advise European governments and enterprises about the need for transparent disclosure of technical information from all open source vendors.

Initiative by the US Department of Commerce

In March 2015, the US Department of Commerce's Internet Policy Task Force (IPTF) announced an initiative to 'address key cybersecurity issues facing the digital economy that could be best addressed by a consensus-based multistakeholder process.'[237] Acknowledging that information technology systems will never be entirely secure, various stakeholders – vendors, regulators, and security researchers – have increasingly sought to promote coordination. This vulnerability coordination discussion is one of such attempts. It is hoped that by opening up the dialogue concerning various aspects of disclosure, participants may come to an agreed set of principles on which future policies can rely.

To prepare for the multistakeholder processes, the IPTF asked participants to prioritise various identified cybersecurity issues which appear in need of coordinated action. Among the issues were botnet mitigation, vulnerability disclosures, consumer security, and distribution of patches. Comments were also requested on methods and structure of the envisioned processes. Particularly notable were the recommendations by the Telecommunication Industry Association (TIA), which stated that 'the most effective solution to ensuring innovation in cybersecurity solutions is to rely on voluntary use of internationally-accepted standards and best practices.'[238] Further, stakeholders emphasised that IPTF should avoid any redundancy by building on the existing body of cybersecurity policies and initiatives.

Given the clear suggestions from the participants, it appears unlikely that the IPTF would on the contrary propose heavier regulations. Besides, the IPTF has previously argued that the traditional regulatory approach was counterproductive in an ever-evolving digital landscape. While the nature of the solutions that may arise from the multi-stakeholder meetings remains largely unpredictable, increasing participation and dialogue by stakeholders is expected to shape an inclusive perspective on cybersecurity.

Some interviewees also emphasised the necessity to facilitate openness and transparency about vulnerability information. Examples may include information sharing platforms such as ISACs, where there is a particular level of trust among the members to share information about vulnerabilities. To an extent, therefore, this already occurs across certain platforms. To facilitate such openness, however, there has to be a sufficient basis for trust as well as a legal commitment from members to respect the non-disclosure aspect of receiving such information.

There is also the necessity of trust building across different communities. Researchers need to be able to trust vendors and vice versa, since such trust is the bedrock of being able to cooperate in a manner which serves the greater good of information security.

---

[237] Simpson, Angela. 2015. 'Enhancing the digital economy through collaboration on vulnerability research disclosure.' National Telecommunications & Information Administration. 9 July. As of 17 August 2015: http://www.ntia.doc.gov/blog/2015/enhancing-digital-economy-through-collaboration-vulnerability-research-disclosure
[238] Thompson, Brian. 2015. 'Commerce Dept. reviewing stakeholder's cybersecurity comments.' *Policy and Regulatory Positioning.* 20 July. As of 17 August 2015: http://www.privsecblog.com/2015/07/articles/policy-regulatory-positioning/commerce-dept-reviewing-stakeholders-cybersecurity-comments/

To facilitate this, a public debate needs to be had about how society wants to deal with information about vulnerabilities. This has become a problem which inherently involves a myriad of stakeholders, and an educated decision about to deal with the challenges is needed.

7. Conclusions

The discussion and debate on vulnerability disclosure has been around for many years. As one respondent described, '*We are approaching the 27th anniversary of the first "Internet Worm" (Nov, 1988), and yet the general form and content of the disclosure debate has changed very little in this time. The fact that as a community we are still unable to establish a consensus on vulnerability disclosure best practices suggests that this is an intrinsically difficult problem*.' This difficulty is in large part the result of different interests held by stakeholders within the vulnerability landscape. The introduction of new players has arguably complicated the dynamics even further. New players include vendors such as car manufacturers, medical device manufacturers and others who have only recently entered the information security scene as their products become increasingly connected to the digital world. So as more mature stakeholders continue to battle the fundamental differences they have among each other, they are also challenged by newcomers to share their knowledge in order to develop a more equal level playing field.

Other aspects of an evolving landscape further complicate the vulnerability ecosystem. The media have begun to play a larger role, which marketing departments of companies have started to cater to in the last couple of years. Vulnerabilities now have catchy names and logos which may overshadow the severity of the vulnerability – or lack thereof – and as such exaggerate vulnerabilities leading to confusion among users. This may create panic among the public and require vendors to devote more resources than necessary. Yet the media also play an essential role in agenda setting when it comes to information security in general and vulnerability disclosure in particular. Without their reporting, certain vendors may never devote the attention to the topic that is required, or users may never find out about the existence of a vulnerability as well as the way to resolve it. The key, however, is to come to a more nuanced reporting of developments where information dissemination plays a more important role than drawing readers' attention with snappy headlines.

Various initiatives, including this project initiated by ENISA, demonstrate the need to bring different stakeholders together to discuss the challenges associated with vulnerability disclosure and the ways such challenges can be addressed. The development of a core set of principles upon which different stakeholders can agree, and to which they can adhere, can go a long way towards reconciling the existence of distinct and at times conflicting interests. To facilitate these discussions and build trust, third parties can play an integral role. In this sense, ENISA can potentially use its standing in the community as well as its expertise in the area of information security to foster connections and facilitate trust building by reaching out not only to vendors and to CSIRTs but also to the researcher community.

Even though vulnerability reward programmes are generally perceived as a positive development that has brought discoverers and vendors closer together, the market must be approached with caution as it can over-incentivise the search and lead to a flood of vulnerabilities, potentially diverting attention and resources away from the most critical challenges.

To conclude, one of the primary challenges to focus upon, and the primary recommendation to put forward with respect to policy development, is the need for an advanced legal landscape to ensure that vulnerability reporting is not endangered by the unintended consequences of criminal and civil legislation. A critical evaluation of the legal landscape, both in terms of criminal law as well as copyright legislation, is needed to ensure security research is appropriately facilitated rather than inappropriately obstructed.

8. References and bibliography

Ablon, Lillian, Libicki, Martin C. Golay Andrea A. 2014. 'Markets for Cybercrime Tools and Stolen Data: Hackers' Bazaar'. Santa Monica, Calif.: RAND Corporation. As of 15 October 2015: http://www.rand.org/content/dam/rand/pubs/research_reports/RR600/RR610/RAND_RR610.pdf

Apple. 2014. 'OS X bash Update 1.0 – OS X Mavericks.' As of 15 October 2015: https://support.apple.com/kb/DL1769?viewlocale=en_US&locale=en_US

Apple. n.d. 'Apple Product Security.' As of 15 October 2015: https://www.apple.com/support/security/

AppSec Consulting. 2015. 'Zero-day Attacks in 2014.' As of 15 October 2015: https://www.appsecconsulting.com/blog/zero-day-attacks-in-2014.

Baggett, Mark. 2014. 'SANS Python Pen Testers | Exploit Heartbleed Vulnerabilities | SEC573.' SANS Penetration Testing, 16 April. As of 15 October 2015 : http://pen-testing.sans.org/blog/2014/04/16/sans-python-pen-testers-exploit-heartbleed-vulnerabilities-sec573-2

Barnes, Richard. 2014. 'The POODLE Attack and the End of SSL 3.0.' Mozilla Security Blog, 14 October. As of 15 October 2015: https://blog.mozilla.org/security/2014/10/14/the-poodle-attack-and-the-end-of-ssl-3-0/

Barrett, Ross. 2014. 'Microsoft patches SandWorm 0-day.' Net-Security, 14 October. As of 15 October 2015: http://www.net-security.org/secworld.php?id=17492

Biancuzzi, Federico. 2008. 'The Laws of Full Disclosure.' Security Focus, 26 February. As of 15 October 2015: http://www.securityfocus.com/columnists/466/2

Bisht, Virendra & William Gamazo Sanchez. 2014. 'Shellshock vulnerabilities proliferate, affect more protocols.' Trend Micro, 2 October. As of 15 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/shellshock-vulnerabilities-proliferate-affect-more-protocols/

Bohme, Rainer. n.d. 'A Comparison of Market Approaches to Software Vulnerability Disclosure.' As of 15 October 2015: https://www.is.uni-muenster.de/security/publications/Boehme2006_CompVulnMarkets_ETRICS.pdf

Brian, Smith. 2014. '[TLS] POODLE applicability to TLS 1.0+ (was Re: Working Group Last Call for draft-ietf-tls-downgrade-scsv-00).' *Internet Engineering Task Force*. 17 October. As of 15 October 2015: https://www.ietf.org/mail-archive/web/tls/current/msg14058.html

Bugcrowd. 2015. 'Standard Disclosure Terms.' As of 15 October 2015: https://bugcrowd.com/resources/standard-disclosure-terms

Facebook. 2015. 'Whitehat.' As of 15 October 2015: https://www.facebook.com/whitehat

Caswell, Matt. 'Forthcoming OpenSSL releases.' Message to openssl-announce mailing list. 15 October 2015. Email: http://marc.info/?l=openssl-announce&m=142653572011212&w=2

Cavusoglu, Hasan, Huseyin Cavusoglu & Raghunathan Srinivasan. n.d. 'Emerging Issues in Responsible Vulnerability Disclosure.' As of 15 October 2015: http://www.infosecon.net/workshop/pdf/65.pdf

Cencini, Andrew, Kevin Yu, and Tony Chan. 2005. 'Software Vulnerabilities: Full-, Responsible-, and Non-Disclosure.' As of 15 October 2015: http://courses.cs.washington.edu/courses/csep590/05au/whitepaper_turnin/software_vulnerabilities_by_cencini_yu_chan.pdf

CERT/CC. n.d. 'Vulnerability Disclosure Policy.' As of 15 October 2015: http://www.cert.org/vulnerability-analysis/vul-disclosure.cfm?

CERT-UK. 2014. 'UPDATE: Bash Vulnerability AKA SHELLSHOCK.' CERT-UK, 9 October. As of 15 October 2015: https://www.cert.gov.uk/resources/advisories/update-bash-vulnerability-aka-shellshock/

Chambers, John T. & John W Thompson. 2004. 'Vulnerability Disclosure Framework.' As of 15 October 2015: http://www.dhs.gov/xlibrary/assets/vdwgreport.pdf

Chang, Ziv. 2014. 'POODLE More Potent, Now Affects TLS.' Trendmicro Security Intelligence Blog, 10 December. As of 15 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/poodle-more-potent-now-affects-tls/

Chazelas, Stephane. 2014, 3 October. 'RE: Shellshock Timeline.' Message to David A Wheeler. Email. As of 15 October 2015: http://seclists.org/oss-sec/2014/q4/92

Chazelas, Stephane. 2014, 8 October. 'How *DID* you find Shellshock?.' Message to David A Wheeler. Email. As of 15 October 2015: http://www.openwall.com/lists/oss-security/2014/10/08/17

Christey, Steve & Chris Wysopal. 2002. 'Responsible Vulnerability Disclosure.' As of 15 October 2015: https://tools.ietf.org/html/draft-christey-wysopal-vuln-disclosure-00#page-3

Cisco. 2014. 'SSL Padding Oracle On Downgraded Legacy Encryption (POODLE) Vulnerability.' As of 15 October 2015: http://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20141015-poodle

Cisco. n.d. 'Security Vulnerability Policy.' As of 15 October 2015: http://www.cisco.com/web/about/security/psirt/security_vulnerability_policy.html

Cloudfare. n.d. 'CloudFlare vulnerability disclosure policy.' As of 15 October 2015: https://www.cloudflare.com/disclosure

Cobalt. 2014. 'The History of Bug Bounty Programs.' Cobalt, 11 April. As of 15 October 2015: https://cobalt.io/blog/the-history-of-bug-bounty-programs/

Committee on National Security Systems (CNSS). 2010. 'National Information Assurance (IA) Glossary.' As of 15 October2015: www.ncsc.gov/publications/policy/docs/CNSSI_4009.pdf

Computer Incidence Response Center Luxemburg (CIRCL). 'Responsible Vulnerability Disclosure.' As of 15 October 2015: https://www.circl.lu/pub/responsible-vulnerability-disclosure/

CVE Details. n.d. 'CVE and CCE Statistics Query Page.' As of 15 October 2015: http://www.cvedetails.com/cvss-score-charts.php?fromform=1&vendor_id=&product_id=&startdate=2001-01-01&enddate=2015-12-31&groupbyyear=1

CVE. 2013. 'Terminology.' As of 15 October 2015: https://cve.mitre.org/about/terminology.html

CVE. 2015. 'About CVE.' As of 15 October 2015: https://cve.mitre.org/about/index.html

Daniel, Michael. 2014. 'Heartbleed: Understanding When We Disclose Cyber Vulnerabilities.' The White House Blog, 28 April. As of 15 October 2015: https://www.whitehouse.gov/blog/2014/04/28/heartbleed-understanding-when-we-disclose-cyber-vulnerabilities

De Muynck, Jo, Hans Graux & Neil Robinson. 2013. 'The Directive on attacks against information systems: A Good Practice Collection for CERTs on the Directive on attacks against information systems.' ENISA. As of 15 October 2015: https://www.enisa.europa.eu/activities/cert/support/fight-against-cybercrime/the-directive-on-attacks-against-information-systems/at_download/fullReport

Donohue, Brian. 2014. 'IBM: Heartbleed Attacks Thousands of Servers Daily.' Threat Post, 27 August. As of 15 October 2015: https://threatpost.com/ibm-heartbleed-attacks-thousands-of-servers-daily/107936#sthash.9qB6TNdR.dpuf

Electronic Foundation Frontier (EFF). 2014. 'EFF Sues NSA, Director of National Intelligence for Zero Day Disclosure Process.' EFF, 1 July. As of 15 October 2015: https://www.eff.org/press/releases/eff-sues-nsa-director-national-intelligence-zero-day-disclosure-process

Electronic Frontier Foundation (EFF). n.d. 'Coders' Rights Project Reverse Engineering FAQ.' As of 15 October 2015: https://www.eff.org/issues/coders/reverse-engineering-faq#faq4

Ellis, Jen. 2015. 'How Do We De-Criminalize Security Research? AKA What's Next for the CFAA?.' Rapid 7 Community, 26 January. As of 15 October 2015: https://community.rapid7.com/community/infosec/blog/2015/01/26/how-do-we-de-criminalize-security-research-aka-what-s-next-for-the-cfaa

ENISA. n.d. 'Glossary.' As of 15 October 2015: http://www.enisa.europa.eu/activities/risk-management/current-risk/risk-management-inventory/glossary#G52

Errata Security. 2015. 'A Call for Better Vulnerability Response.' As of 15 October 2015: http://blog.erratasec.com/2015/01/a-call-for-better-vulnerability-response.html#.VWh4ms9VhBe

European Commission. 2014. 'Commission updates EU control list on dual use items.' Trade, Dual Use Control, 22 October. As of 15 October 2015: http://trade.ec.europa.eu/doclib/press/index.cfm?id=1166&title=Commission-updates-EU-control-list-on-dual-use-items

Evans, Chris & Hintz, Drew, 2013. 'Disclosure timeline for vulnerabilities under active attack.' Google Online Security Blog, 29 May. As of 15 October 2015: http://googleonlinesecurity.blogspot.com.au/2013/05/disclosure-timeline-for-vulnerabilities.html

Evans, Chris, Hawkes, Ben, Adkins, Heather, Moore, Matt, Zalewski, Michal & Gerhard Eschelbeck. 2015. 'Feedback and data-driven updates to Google's disclosure policy.' Project Zero, 13 February. As of 15 October 2015: http://googleprojectzero.blogspot.com/2015/02/feedback-and-data-driven-updates-to.html

Facebook Bug Bounty. n.d. In *Facebook* Product/Service page. As of 15 October 2015: https://www.facebook.com/BugBounty

Facebook. 2015. 'Whitehat.' As of 15 October 2015: https://www.facebook.com/whitehat

Fedora Update Notification. 2014. '[SECURITY] Fedora 21 Update: bash-4.3.25-2.fc21.' As of 15 October 2015: https://lists.fedoraproject.org/pipermail/package-announce/2014-September/139129.html

Finkle, Jim. 2014. 'New Poodle web threat not seen as menacing as Heartbleed, Shellshock.' Reuters, 15 October. As of 15 October 2015: http://in.reuters.com/article/2014/10/15/cybersecurity-encryption-poodle-idINKCN0I401X20141015

Francis, Hannah. 2014. 'Shellshock: The latest security superbug explained.' The Sydney Morning Herald, 27 September. As of 15 October 2015: http://www.smh.com.au/digital-life/consumer-security/shellshock-the-latest-security-superbug-explained-20140927-10mcfx.html

F-Secure. 2014. 'Blackenergy & Quedagh – The Convergence of crimeware and APT attacks.' As of 15 October 2015:
https://www.f-secure.com/documents/996508/1030745/blackenergy_whitepaper.pdf

Geftic, Seth. 2014. 'Patching Poodles and Digging for Sandworms: Why Monitoring Matters.' RSA Security Operations Blog, 17October. As of 15 October 2015: https://blogs.rsa.com/patching-poodles-digging-sandworms-monitoring-matters/

Global Conference on Cyberspace 2015. n.d. 'Global Forum On Cyber Expertise (GFCE).' As of 15 October 2015: https://www.gccs2015.com/gfce

Glyer, Christopher. 2014. 'Attackers Exploit the Heartbleed OpenSSL Vulnerability to Circumvent Multi-factor Authentication on VPNs.' FireEye, 18 April. As of 15 October 2015: https://www.mandiant.com/blog/attackers-exploit-heartbleed-openssl-vulnerability-circumvent-multifactor-authentication-vpns/#sthash.WpNMsglj.dpuf

Google. n.d. 'Google Application Security.' As of 15 October 2015: https://www.google.com/about/appsecurity/

Greenberg, Andy. 2012. 'Meet the Hackers Who Sell Spies the Tools to Crack Your PC (And Get Paid Six-Figure Fees).' Forbes, 21 March. As of 7 October 2015: http://www.forbes.com/sites/andygreenberg/2012/03/21/meet-the-hackers-who-sell-spies-the-tools-to-crack-your-pc-and-get-paid-six-figure-fees/

Grubb, Ben. 2014. 'Google accused of being selfish and playing favourites over Heartbleed security bug disclosure.' *The Age*, 19 April. As of 15 October 2015: http://www.theage.com.au/it-pro/security-it/google-accused-of-being-selfish-and-playing-favourites-over-heartbleed-security-bug-disclosure-20140418-zqvvk.html

Heartbleed.com. 2014. 'The Heartbleed Bug.' As of 15 October 2015: http://heartbleed.com/

HackerOne. n.d. As of 15 October 2015: https://hackerone.com/

Hill, Kashmir. 2014. 'Apple's Deafening Silence on 'GoToFail' Security Flaw.' Forbes, 24 February. As of 30 September 2015: http://www.forbes.com/sites/kashmirhill/2014/02/24/apples-deafening-silence-on-gotofail-security-flaw/

Hofmann, Marcia. 2012. '"Zero-day" exploit sales should be key point in cybersecurity debate.' Electronic Frontier Foundation (EFF), 29 March. As of 15 October 2015: https://www.eff.org/deeplinks/2012/03/zero-day-exploit-sales-should-be-key-point-cybersecurity-debate

Hultquist, John. 2014. 'Sandworm Team – Targeting SCADA Systems.' iSIGHT Partners Blog, 21October. As of 15 October 2015: http://www.isightpartners.com/2014/10/sandworm-team-targeting-scada-systems/

ICASI. 2015. 'Current Activities.' As of 15 October 2015: http://www.icasi.org/current-activities/

InfoCuria – Case –Law of the Court of Justice. 2012. 'JUDGMENT OF THE COURT (Grand Chamber).' As of 15 October 2015:
http://curia.europa.eu/juris/document/document.jsf?text=&docid=122362&pageIndex=0&doclang=en&mode=req&dir=&occ=first&part=1&cid=564907

Internet Bug Bounty. n.d. 'Internet Bug Bounty: Rewarding friendly hackers who contribute to a more secure internet.' As of 15 October 2015: https://hackerone.com/internet-bug-bounty

ISO. 2013. 'Information technology -- Security techniques -- Vulnerability handling processes.' As of 15 October 2015: http://www.iso.org/iso/catalogue_detail.htm?csnumber=53231

ISO. 2014. 'Information technology -- Security techniques -- Vulnerability disclosure.' As of 15 October: 2015: http://www.iso.org/iso/catalogue_detail.htm?csnumber=45170

Itnews. 2007. 'Vista contest offers cash for exploits.' As of 25 October: http://www.itnews.com.au/news/vista-contest-offers-cash-for-exploits-71238

Jackson, Joab. 2014. 'Microsoft Patch Tuesday tackles three critical vulnerabilities, including 'Sandworm'.' PC World, 14 October. As of 15 October 2015: http://www.pcworld.com/article/2833852/microsoft-patch-tuesday-tackles-three-critical-vulnerabilities.html

Kahl, Chad. 2014a. 'The Shellshock BaSH Bug: Vulnerability in BaSH is a Big Deal.' Solutionary, 25 September. As of 15 October 2015: http://www.solutionary.com/resource-center/blog/2014/09/shellshock-vulnerability-in-bash-is-a-big-deal/

Kahl, Chad. 2014b. 'Shellshock: Accelerating The Standard Timeline.' Solutionary, 26 September. As of 15 October 2015: http://www.solutionary.com/resource-center/blog/2014/09/shellshock-accelerating-the-standard-timeline/

Kaminsky, Dan. 2014. 'Be Still My Breaking Heart.' Dan Kaminsky's Blog, 10 April. As of 15 October 2015: http://dankaminsky.com/2014/04/10/heartbleed/

Kirsch, Cassandra. n.d. 'The Grey Hat Hacker: Reconciling Cyberspace Reality and the Law.' As of 15 October 2015: http://www.nku.edu/content/dam/chaselaw/docs/centersandinstitute/Law---Informatics/Symposium-CLE/Kirsch_Working%20Paper_The%20Grey%20Hat%20Hacker.pdf

Kovacs, Eduard. 2014. 'Several Siemens Industrial Products Affected by ShellShock Bug.' Security Week, 8 October. As of 15 October 2015: http://www.securityweek.com/several-siemens-industrial-products-affected-shellshock-bug

Krebs On Security. n.d. 'Posts Tagged: Heartbleed.' As of 15 October 2015: http://krebsonsecurity.com/tag/heartbleed/

Langley, Adam. 2014. 'The POODLE bites again.' Imperialviolet.org, 8 December. As of 15 October 2015: https://www.imperialviolet.org/2014/12/08/poodleagain.html

Lee, Dave. 2014. 'Shellshock: 'Deadly serious' new vulnerability found.' BBC, 25 September. As of 15 October 2015: http://www.bbc.com/news/technology-29361794

Lenovo. 2015. 'Lenovo Vulnerability Disclosure Policy.' As of 15 October 2015: https://support.lenovo.com/us/en/documents/ht103338

Lindh, Andreas (addelindh). '@markstanislav Shell schock.' 24 September 2014, 9:42. As of 15 October 2015: https://mobile.twitter.com/addelindh/status/514817121101283328

Marquess, Steve. 2014. 'Of Money, Responsibility, and Pride.' SPEEDS AND FEEDS, 12 April. As of 15 October 2015: http://veridicalsystems.com/blog/of-money-responsibility-and-pride/

Microsoft Developer Network. n.d. 'Definition of a Security Vulnerability.' As of 9 August 2015: https://msdn.microsoft.com/en-us/library/Cc751383.aspx

Microsoft Security Bulletin MS14-064. 2014. 'Vulnerabilities in Windows OLE Could Allow Remote Code Execution (3011443).' Security TechCenter, 11 November. As of 15 October 2015: https://technet.microsoft.com/library/security/MS14-064

Microsoft. 2011. 'Coordinated Vulnerability Disclosure at Microsoft.' As of 15 October 2015: http://go.microsoft.com/?linkid=9770197

Mimoso, Michael. 2014. 'Researcher takes wraps off two undisclosed shellshock vulnerabilities in Bash.' Threatpost, 3 October. As of 15 October 2015: https://threatpost.com/researcher-takes-wraps-off-two-undisclosed-shellshock-vulnerabilities-in-bash/108674/

Möller, Bodo, Thai Duong & Kotowicz, Krzysztof. 2014. 'Security Advisory - This POODLE Bites: Exploiting the SSL 3.0 Fallback.' As of 15 October 2015: https://www.openssl.org/~bodo/ssl-poodle.pdf

Möller, Bodo. 2014. 'This POODLE bites: exploiting the SSL 3.0 fallback.' Google Online Security Blog, 14 October. As of 15 October 2015: http://googleonlinesecurity.blogspot.be/2014/10/this-poodle-bites-exploiting-ssl-30.html

Moussouris, Katie. 2015. 'Vulnerability Disclosure Deja Vu: Prosecute Crime Not Research.' *DarkReading*, 12May. As of 15 October 2015: http://www.darkreading.com/vulnerabilities---threats/vulnerability-disclosure-deja-vu-prosecute-crime-not-research/a/d-id/1320384

Mozilla Security. n.d. 'Bug bounty program.' As of 15 October 2015: https://www.mozilla.org/en-US/security/bug-bounty/

National Cyber Security Centre (NCSC). 2013a. *Policy for Arriving at a Practice forRresponsible Disclosure*. As of 15 October 2015: https://www.ncsc.nl/binaries/content/documents/ncsc-en/current-topics/news/responsible-disclosure-guideline/1/Responsible%2BDisclosure%2BENG.pdf

National Institute of Standards and Technology. n.d. – a. ‘National Vulnerability Database.’ As of 15 October 2015: https://nvd.nist.gov/

National Institute of Standards and Technology. n.d.-b. ‘CVSS v3 Information: NVD Common Vulnerability Scoring System Support v2.’ As of 15 October 2015: https://nvd.nist.gov/CVSS.aspx

National Institute of Standards and Technology. n.d.-c. ‘CVE and CCE Statistics Query Page.’ As of 15 October 2015: https://web.nvd.nist.gov/view/vuln/statistics

National Institute of Standards and Technology. 2015a. ‘Vulnerability Summary for CVE-2014-0160.’ As of 15 October 2105: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-0160]

National Institute of Standards and Technology. 2015b. ‘Vulnerability Summary for CVE-2014-4114.’ As of 15 October 2105: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-4114]

National Institute of Standards and Technology. 2015c. ‘Vulnerability Summary for CVE-2014-6271.’ As of 15 October 2105: http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-6271]

National Institute of Standards and Technology. 2015d. ‘Vulnerability Summary for CVE-2014-3566.’ As of 15 October 2105: http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-3566]

NCSC. 2013b. ‘Responsible Disclosure Guideline.’ As of 15 October 2015: https://www.ncsc.nl/english/current-topics/news/responsible-disclosure-guideline.html

NCSC. 2015. ‘Introducing Responsible Disclosure: Experiences in the Netherlands: A Best Practice Guide.’ As of 15 October 2015: https://www.gccs2015.com/sites/default/files/documents/BestPracticeRD-20150409_0.pdf

National Institute of Standards and Technology (NIST). 2010. ‘Guide for Applying the Risk Management Framework to Federal Information Systems: A Security Life Cycle Approach.’ As of 15 August 2015: http://www.csrc.nist.gov/publications/nistpubs/800-37-rev1/sp800-37-rev1-final.pdf

Network Working Group. 2000. ‘Internet Security Glossary.’ As of 15 October 2015: https://tools.ietf.org/html/rfc2828

Nolette, Ryan. 2014. ‘After Taking a Bite Out of SSL 3.0, This POODLE Needs Some Time in Obedience Class.’ Bit9 Blog, 15 October. As of 15 October 2015: https://blog.bit9.com/2014/10/15/after-taking-a-bite-out-of-ssl-3-0-this-poodle-needs-some-time-in-obedience-class/

Official Journal of the European Union. 2014, 30 December. ‘Legislation.’ Vol. 57. As of 15 October 2015: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=OJ:L:2014:371:FULL&from=EN

Open SUSE Security. n.d. ‘Mailinglist Archive: opensuse-security-announce (44 mails).’ As of 15 October 2015: http://lists.opensuse.org/opensuse-security-announce/2014-09/msg00042.html

Openbaar Ministerie. 2013a. 'Beleid OM 'ethische hackers' in lijn met 'leidraad Responsible Disclosure.' As of 15 October 2015: https://www.om.nl/actueel/nieuwsberichten/@32028/beleid-ethische/

Openbaar Ministerie. 2013b, 18 March. Letter to: Aan alle parkethoofden. As of 15 October 2015: https://www.om.nl/publish/pages/22742/03_18_13_beleidsbrief_college_responsible_disclosure.pdf

Oracle. n.d. 'Oracle Security Vulnerability Disclosure Policies.' As of 15 October 2015: http://www.oracle.com/us/support/assurance/vulnerability-remediation/reporting-security-vulnerabilities/index.html

Organization for Internet Safety. 2004. 'Guidelines for Security Vulnerability Reporting and Response.' As of 15 October 2015: https://www.symantec.com/security/OIS_Guidelines%20for%20responsible%20disclosure.pdf

Paganini, Pierluigi. 2014. 'POODLE SSL flaw is threatening also TLS Security Protocol.' Security Affairs, 10 December. As of 15 October 2015: http://securityaffairs.co/wordpress/30952/hacking/poodle-tls-flaw.html

Perlroth, Nicole. 2014. 'Security Experts Expect 'Shellshock' Software Bug in Bash to Be Significant.' New York Times, 25 September. As of 15 October 2015: http://www.nytimes.com/2014/09/26/technology/security-experts-expect-shellshock-software-bug-to-be-significant.html?_r=3

Perlroth, Nicole. 2015. 'HackerOne Connects Hackers With Companies, and Hopes for a Win-Win.' New York Times, 7 June. As of 15 October 2015: http://www.nytimes.com/2015/06/08/technology/hackerone-connects-hackers-with-companies-and-hopes-for-a-win-win.html?_r=0

Pettersen, Yngve Nysaeter. 2014. 'Not out of the woods yet: There are more POODLEs.' Vivaldi Blog, 9 December. As of 8 October 2015: https://vivaldi.net/userblogs/entry/not-out-of-the-woods-yet-there-are-more-poodles

Rainie, Lee & Maeve Duggan. 2014. 'Heartbleed's Impact.' Pew Research Centre, 30 April. As of 15 October 2015: http://www.pewinternet.org/2014/04/30/heartbleeds-impact/

Ramey, Chet. n.d. 'BASH - The Bourne-Again Shell.' As of 15 October 2015: http://tiswww.case.edu/php/chet/bash/bash-intro.html]

Rapid7. 2015. 'Statement for Hearing on Cyber Crime: Modernizing our Legal Framework for the Information Age.' As of 15 October 2015: http://www.judiciary.senate.gov/imo/media/doc/07-08-15%20Ellis%20Testimony.pdf

Rapid7. n.d. 'Vulnerability Disclosure Policy.' As of 15 October 2015: https://www.rapid7.com/disclosure.jsp

Red Hat Product Security. 2014. 'Can SSL 3.0 be fixed? An analysis of the POODLE attack.' Security Blog, October 20. As of 15 October 2015: https://securityblog.redhat.com/2014/10/20/can-ssl-3-0-be-fixed-an-analysis-of-the-poodle-attack/

Red Hat. 2014a, September 26. 'Important: bash security update.' As of 15 October 2015: https://rhn.redhat.com/errata/RHSA-2014-1306.html

Red Hat. 2014b, October 2. 'Bash Code Injection Vulnerability via Specially Crafted Environment Variables (CVE-2014-6271, CVE-2014-7169).' As of 15 October 2015: https://access.redhat.com/articles/1200223

Ring, Tim. 2014. 'Tens of thousands of servers *still* vulnerable to Heartbleed.' SC Magazine, 9 May. As of 15 October 2015 : http://www.scmagazineuk.com/tens-of-thousands-of-servers-still-vulnerable-to-heartbleed/article/346268/

Ristic, Ivan. 2014. 'Poodle Bites TLS.' Qualys Blog, Security Labs, December 8. As of 8 October 2015: https://community.qualys.com/blogs/securitylabs/2014/12/08/poodle-bites-tls

Saarinen, Juha. 2014. 'First Shellshock botnet attacks Akamai, US DoD networks.' IT News, 26 September. As of 15 October 2015: http://www.itnews.com.au/News/396197,first-shellshock-botnet-attacks-akamai-us-dod-networks.aspx

Sapio, Tim. 2014. 'Heartbleed: Examining The Impact.' DarkReading, 10 April. As of 15 October 2015: http://www.darkreading.com/attacks-breaches/heartbleed-examining-the-impact-/d/d-id/1204330

Sass, Jeff. 2015. 'The Role of Static Analysis in Heartbleed.' SANS Institute. As of 15 October 2015: http://www.sans.org/reading-room/whitepapers/threats/role-static-analysis-heartbleed-35752

Schiffman Mike. 2011. 'The Common Vulnerability Reporting Framework: An Internet Consortium for Advancement of Security on the Internet (ICASI) Whitepaper.' As of 15 October 2015: http://www.icasi.org/wp-content/uploads/2015/06/cvrf-whitepaper.pdf

Schindler, Helen R., Jonathan Cave, Neil Robinson, Veronika Horvath, Petal Hackett, Salil Gunashekar, Maarten Botterman, Simon Forge and Hans Graux. 2012. Europe's policy options for a dynamic and trustworthy development of the Internet of Things. Santa Monica: Calif.: RAND Corporation. As of 12 October 2015: http://www.rand.org/content/dam/rand/pubs/research_reports/RR300/RR356/RAND_RR356.pdf

Schneier, Bruce. 2007. 'Schneier: Full Disclosure of Security Vulnerabilities a 'Damned Good Idea'.' As of 15 October 2015: https://www.schneier.com/essays/archives/2007/01/schneier_full_disclo.html

Schneier, Bruce. 2012. 'The Vulnerabilities Market and the Future of Security.' Forbes, May 30. As of 7 October 2015 : http://www.forbes.com/sites/bruceschneier/2012/05/30/the-vulnerabilities-market-and-the-future-of-security/

Scott, Rob. 2014. 'Security Experts Warn of Potentially Dangerous Shellshock Bug .' ETCentric, 26 September. As of 15 October 2015: http://www.etcentric.org/security-experts-warn-of-potentially-dangerous-shellshock-bug

Shannon_Sabens. 2015. 'Milestone today, good times ahead.' HP Security Research Blog, 12 May. As of 15 October 2015: http://h30499.www3.hp.com/t5/HP-Security-Research-Blog/Milestone-today-good-times-ahead/ba-p/6743824#.VWmwrU0tGUk

Simpson, Angela. 2015. 'Enhancing the digital economy through collaboration on vulnerability research disclosure.' National Telecommunications & Information Administration. 9 July 9. As of 15 October 2015: http://www.ntia.doc.gov/blog/2015/enhancing-digital-economy-through-collaboration-vulnerability-research-disclosure

Stafford, Jared. 2014. 'http://s3.jspenguin.org/ssltest.py.' Pastebin, 8 April. As of 15 October 2015 : http://pastebin.com/WmxzjkXJ

Stallman, Erik. 2015. 'The Current DMCA Exemption Process is a Computer Security Vulnerability.' Centre for Democracy and Technology, 21 January. As of 15 October 2015: https://cdt.org/blog/the-current-dmca-exemption-process-is-a-computer-security-vulnerability/

Symantec. 2014. 'Sandworm Windows zero-day vulnerability being actively exploited in targeted attacks.' Symantec Security Response Blog, 14 October. As of 15 October 2015: http://www.symantec.com/connect/blogs/sandworm-windows-zero-day-vulnerability-being-actively-exploited-targeted-attacks

Symantec. n.d.-a. 'Vulnerability Trends.' As of 9 August 2015: https://securityresponse.symantec.com/en/uk/threatreport/topic.jsp?id=vulnerability_trends

Symantec. n.d.-b. 'Vulnerability Management Commitment and Disclosure Policy.' As of 15 October 2015: https://www.symantec.com/security/

Techopedia. n.d. 'Vulnerability.' As of 15 October 2015: http://www.techopedia.com/definition/13484/vulnerability

The Economist. 2014. 'Ghosts in the machine language.' The Economist, 24 October. As of 15 October 2015: http://www.economist.com/news/science-and-technology/21627868-latest-hacks-and-exploits-result-benign-neglect-and-wont-be-last-ghosts-machine

Thompson, Brian. 2015. 'Commerce Dept. reviewing stakeholder's cybersecurity comments.' Policy and Regulatory Positioning*. 20* July. As of 15 October 2015: http://www.privsecblog.com/2015/07/articles/policy-regulatory-positioning/commerce-dept-reviewing-stakeholders-cybersecurity-comments/

Tran, Cindee. 2015. 'Zero-day Attacks in 2014.' App Sec Consulting, 16 January. As of 15 October 2015: https://www.appsecconsulting.com/blog/zero-day-attacks-in-2014

Trend Micro. 2014., 'Shellshock: A Technical Report.' As of 15 October 2015: http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-shellshock.pdf

Ubuntu. 2014, 27 September. 'USN-2364-1: Bash vulnerabilities.' As of 15 October 2015: http://www.ubuntu.com/usn/usn-2364-1/

Uchiyama, Takayuki. 2014. 'Year in Review - Vulnerability Handling and Changing with the Times.' JP CERT, 11 December. As of 15 October 2015: http://blog.jpcert.or.jp/2014/12/year-in-review---vulnerability-handling-and-changing-with-the-times.html

United. n.d. 'United Airlines bug bounty program.' As of 15 October 2015: http://www.united.com/web/en-US/content/contact/bugbounty.aspx

United States Copyright Office. n.d.-a. 'Executive Summary Digital Millennium Copyright Act: Section 104 Report.' As of 15 October 2015: http://www.copyright.gov/reports/studies/dmca/dmca_executive.html

United States Copyright Office. n.d.-b. 'Exemption to Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies Under 17 U.S.C. § 1201.' As of 15 October 2015: http://copyright.gov/1201/2015/post-hearing/answers/Class_25_Hearing_Response_CDT-OTI_Docket_No_2014-07_2015.pdf

Vaughan-Nichols, Steven J. 2015. ‘Shellshock: Better 'bash' patches now available.’ ZD Net, 27 September. As of 15 October 2015: http://www.zdnet.com/article/shellshock-better-bash-patches-now-available/

Ward, Stephen. 2014. ‘iSIGHT discovers zero-day vulnerability CVE-2014-4114 used in Russian cyber-espionage.’ iSIGHT Partners Blog, 14 October. As of 15 October 2015: http://www.isightpartners.com/2014/10/cve-2014-4114/

Wassenaar Arrangement. n.d. ‘Introduction.’ As of 15 October 2015: http://www.wassenaar.org/introduction/index.html

Weimer, Florian. 2014, 24 September. ‘CVE-2014-6271: remote code execution through bash.’ Email. As of 15 October 2015: http://seclists.org/oss-sec/2014/q3/649

Weith, Loren. 2006. ‘Differences between SSLv2, SSLv3, and TLS.’ As of 15 October 2015: http://www.yaksman.org/~lweith/ssl.pdf

Wheeler, David A. 2015. ‘Shellshock.’ Dweeler.com, 13 February. As of 15 October 2015: http://www.dwheeler.com/essays/shellshock.html#timeline

Wilhoit, Kyle & Jim Gogolinski. 2014. ‘Sandworm to Blacken – The SCADA Connection.’ Trendlabs Security Intelligence Blog, 16 October. As of 15 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/sandworm-to-blacken-the-scada-connection/

Wood, Ian. 2014. ‘UK: Court Of Appeal Upholds High Court Decision In SAS Institute Inc V World Programming Ltd Software Infringement Case.’ Mondaq, 14 April 4. As of 15 October 2015: http://www.mondaq.com/x/304618/Copyright/Court+Of+Appeal+Upholds+High+Court+Decision+In+SAS+Institute+Inc+V+World+Programming+Ltd+Software+Infringement+Case

Wu, Weimin. 2014. ‘An Analysis of Windows Zero-Day Vulnerability ‘CVE 2014-4114’ aka Sandworm.’ Trendlabs Security Intelligence Blog, 14 October. As of 15 October 2015: http://blog.trendmicro.com/trendlabs-security-intelligence/an-analysis-of-windows-zero-day-vulnerability-cve-2014-4114-aka-sandworm/

Yahoo. 2014. ‘Users First: Our Vulnerability Disclosure Policy.’ As of 15 October 2015: http://yahoopolicy.tumblr.com/post/104777538533/users-first-our-vulnerability-disclosure-policy

Zero Day Initiative. n.d.-a. ‘Why Did We Create the Zero Day Initiative?.’ As of 15 October 2015: http://www.zerodayinitiative.com/about/

Zero Day Initiative. n.d.-b. ‘Disclosure Policy.’ As of 15 October 2015: http://www.zerodayinitiative.com/advisories/disclosure_policy/

Zetter, Kim. 2015a. ‘United Airlines Pays Man A Million Miles for Reporting Bug.’ Wired, July 14. As of 7 October 2015: http://www.wired.com/2015/07/united-airlines-pays-man-million-miles-reporting-bug/

Zetter, Kim. 2015b. ‘With Lock Research, Another Battle Brews in the War Over Security Roles.’ Wired, 6 May. As of 15 October 2015: http://www.wired.com/2015/05/lock-research-another-battle-brews-war-security-holes/

Zetter, Kim. 2015c. 'A Bizarre Twist in the Debate over Vulnerability Disclosures.' Wired, 11 September. As of 15 October 2015: http://www.wired.com/2015/09/fireeye-enrw-injunction-bizarre-twist-in-the-debate-over-vulnerability-disclosures/

Annex A: Yearly statistics of reported vulnerabilities from the National Vulnerability Database

Table 3: Yearly statistics (2001-2014) showing the number and share of reported vulnerabilities broken down according to their severity 'rating'; the severity ratings are dependent on the computed NVD CVSS score (Source: the graph was produced using CVE data retrieved from the National Vulnerability Database)

Year	Number of 'low' severity vulnerabilities	Number of 'medium' severity vulnerabilities	Number of 'high' severity vulnerabilities	Total number of vulnerabilities	Fraction of 'low' severity vulnerabilities	Fraction of 'medium' severity vulnerabilities	Fraction of 'high' severity vulnerabilities
2001	190	714	773	1,677	11.33%	42.58%	46.09%
2002	153	999	1,004	2,156	7.10%	46.34%	46.57%
2003	100	749	678	1,527	6.55%	49.05%	44.40%
2004	208	1,274	969	2,451	8.49%	51.98%	39.53%
2005	454	2,437	2,040	4,931	9.21%	49.42%	41.37%
2006	515	3,332	2,761	6,608	7.79%	50.42%	41.78%
2007	231	3,125	3,158	6,514	3.55%	47.97%	48.48%
2008	186	2,607	2,839	5,632	3.30%	46.29%	50.41%
2009	199	2,814	2,719	5,732	3.47%	49.09%	47.44%
2010	278	2,267	2,094	4,639	5.99%	48.87%	45.14%
2011	260	2,069	1,821	4,150	6.27%	49.86%	43.88%
2012	511	3,013	1,764	5,288	9.66%	56.98%	33.36%
2013	519	2,930	1,737	5,186	10.01%	56.50%	33.49%
2014	661	5,356	1,920	7,937	8.33%	67.48%	24.19%

Annex B: List of interviewees

B.1 Telephone interviews

1. Anonymous
2. Anonymous
3. Benning, Rob. *ING*
4. Böhme, Rainer. *University of Muenster*
5. Borrett, Martin. *IBM*
6. Cardozo, Nate. *Electronic Frontier Foundation (EFF)*
7. Day, Greg. *FireEye*
8. Ellis, Ryan. *Harvard University*
9. Genes, Raimund. *TrendMicro*
10. Harris, Duncan. *Oracle*
11. Knake, Rob. *Council on Foreign Relations (CFR)*
12. Manion, Art. *CERT-CC*
13. Scheuring, Christopher. *Enno Rey Netzwerke GmbH (ERNW)*
14. Schreck, Thomas. *Siemens*
15. Van der Ham, Jeroen. *Dutch National Cyber Security Centre (NCSC)*
16. Van Horenbeeck, Maarten. *Fastly*

B.2 Written contributions

1. JPCert Coordination Center
2. Rajnovic, Damir. *Cisco Product Security Incident Response Team*
3. Project Zero team. *Google*
4. Olivé Leite, Fábio & Seifried, Kurt. *Red Hat Product Security*

Annex C: Indicative interview protocol

- What is your background pertaining to vulnerabilities and vulnerability disclosure?
- What are currently the three main challenges with vulnerability disclosure?
- What are good practices in the area of vulnerability disclosure?
- What changes would you suggest/like to see?
- How could such recommendations be implemented?
- Who should implement them?
- What role can CSIRTs play in vulnerability disclosure?
- What role can ENISA play in vulnerability disclosure?
- What role do the media play in vulnerability disclosure in your experience?

Annex D: Sample list of advisories and alerts issued in relation to the four vulnerabilities covered in the case studies

Organisation	Vulnerability	Advisory/Alert
United States Computer Emergency Response Team (US-CERT)	Heartbleed	*'A vulnerability in OpenSSL could allow a remote attacker to expose sensitive data, possibly including user authentication credentials and secret keys, through incorrect memory handling in the TLS heartbeat extension. This may allow an attacker to decrypt traffic or perform other attacks. OpenSSL version 1.0.1g resolves this vulnerability. The 1.0.0 and 0.9.8 branches are not vulnerable.* *US-CERT recommends users and administrators review Vulnerability Note VU#720951 for additional information and mitigation details.'* [Source: https://www.us-cert.gov/ncas/current-activity/2014/04/08/OpenSSL-Heartbleed-Vulnerability]
HP	Heartbleed	*'The Heartbleed vulnerability was detected in specific OpenSSL versions. OpenSSL is a 3rd party product that is embedded with some of HP Software products. This bulletin objective is to notify HP Software customers about products affected by the Heartbleed vulnerability... HP Software is working to address this vulnerability for all affected product versions. HP Software will release product specific security bulletins for each impacted product.'* [Source: http://marc.info/?l=bugtraq&m=139722163017074&w=2]
Microsoft	Sandworm	*'A vulnerability exists in Windows OLE that could allow remote code execution if a user opens a file that contains a specially crafted OLE object. An attacker who successfully exploited this vulnerability could gain the same user rights as the logged-on user. If the current user is logged on with administrative user rights, an attacker could then install programs; view, change, or delete data; or create new accounts with full user rights. Customers whose accounts are configured to have fewer user rights on the system could be less impacted than users who operate with administrative user rights.'* [Source: https://technet.microsoft.com/library/security/ms14-060#ID0EQAAE]
iSIGHT Partners	Sandworm	*'On Tuesday, October 14, 2014, iSIGHT Partners – in close collaboration with Microsoft – announced the discovery of a zero-day vulnerability impacting all supported versions of Microsoft Windows and Windows Server 2008 and 2012.* *Microsoft is making a patch for this vulnerability available as part of patch updates on the 14th – CVE-2014-4114.'* [Source: http://www.isightpartners.com/2014/10/cve-2014-4114/]
United States Computer Emergency Response Team (US-CERT)	Shellshock	*'US-CERT is aware of a Bash vulnerability affecting Unix-based operating systems such as Linux and Mac OS X. Exploitation of this vulnerability may allow a remote attacker to execute arbitrary code on an affected system... US-CERT recommends users and administrators review TA14-268A, Vulnerability Note VU#252743 and the Redhat Security Blog (link is external) for additional details and to refer to their respective Linux or Unix-based OS vendor(s) for an appropriate patch. A GNU Bash patch is also available for experienced users and administrators to implement…'* [Source: https://www.us-cert.gov/ncas/current-activity/2014/09/24/Bourne-Again-Shell-Bash-Remote-Code-Execution-Vulnerability]
IBM	Shellshock	*'Six Bash vulnerabilities were disclosed in September 2014. This bulletin addresses the vulnerabilities that have been referred to as "Bash Bug" or "Shellshock" and two memory corruption vulnerabilities. Bash is used by IBM Security Access Manager for Mobile and IBM Security Access Manager for Web…'* [Source: http://www-01.ibm.com/support/docview.wss?uid=swg21685733]
SecurityFocus	POODLE	*'...Impact: An attacker may be able to decrypt data protected by SSL. Description: There are known attacks on the confidentiality of SSL 3.0 when a cipher suite uses a block cipher in CBC mode. An attacker could force the use of SSL 3.0, even when the server would support a better TLS version, by blocking TLS 1.0 and higher connection attempts...'* [Source: http://www.securityfocus.com/archive/1/archive/1/533724/100/0/threaded]
Red Hat	POODLE	*'A flaw was found in the way SSL 3.0 handled padding bytes when decrypting messages encrypted using block ciphers in cipher block chaining (CBC) mode. This flaw allows a man-in-the-middle (MITM) attacker to decrypt a selected byte of a cipher text in as few as 256 tries if they are able to force a victim application to repeatedly send the same data over newly created SSL 3.0 connections.'* [Source: https://access.redhat.com/security/cve/CVE-2014-3566]

Annex E: Vulnerability disclosure policy template

E.1 Vulnerability disclosure policy template

The length and complexity of a vulnerability disclosure policy differs from company to company. Oracle's disclosure policy for instance is just three sentences long,[239] while Microsoft provides an eight page document.[240]

There are however numerous elements that the majority of vulnerability disclosure policies have in common.

E.1.1 Security and disclosure philosophy

The first line of a vulnerability disclosure policy usually highlights a company's security philosophy.

Google for instance starts with: 'As a provider of software and services for many users, advertisers, and publishers on the internet, we recognize how important it is to help protect your privacy and security.'[241] Other examples include, 'Lenovo is committed to delivering safe and secure products and services'[242], and Cloudfare states that, 'we take security, trust and transparency seriously.'[243]

Template:

When it comes to security, our users come first.

The second phrase reiterates a broad statement regarding the company's disclosure philosophy.

Facebook notes that, 'we will investigate all legitimate reports and do our best to quickly fix the problem.'[244] Symantec highlights that it is 'committed to resolving security vulnerabilities quickly and carefully.'[245] And Yahoo clarifies that, 'when we discover previously unknown security vulnerabilities, we immediately address the risks on our own systems to protect our users.'[246]

Template:

[xxx] is committed to fix all reported security vulnerabilities quickly and carefully to protect the security and privacy of our users.

[239] Oracle. n.d. 'Oracle Security Vulnerability Disclosure Policies.' As of 14 September 2015: http://www.oracle.com/us/support/assurance/vulnerability-remediation/reporting-security-vulnerabilities/index.html
[240] Microsoft. 2011. 'Coordinated Vulnerability Disclosure at Microsoft.' As of 14 September 2015: http://go.microsoft.com/?linkid=9770197
[241] Google. n.d. 'Google Application Security.' As of 14 September 2015: https://www.google.com/about/appsecurity/
[242] Lenovo. 2015. 'Lenovo Vulnerability Disclosure Policy.' As of 14 September 2015: https://support.lenovo.com/us/en/documents/ht103338
[243] Cloudfare. n.d. 'CloudFlare vulnerability disclosure policy.' As of 14 September 2015: https://www.cloudflare.com/disclosure
[244] Facebook. 2015. 'Whitehat.' As of 14 September 2015: https://www.facebook.com/whitehat
[245] Symantec. n.d. 'Vulnerability Management Commitment and Disclosure Policy.' As of 14 September 2015: https://www.symantec.com/security/
[246] Yahoo. 2014. 'Users First: Our Vulnerability Disclosure Policy.' As of 14 September 2015: http://yahoopolicy.tumblr.com/post/104777538533/users-first-our-vulnerability-disclosure-policy

E.1.2 Reporting the vulnerability

This paragraph ought to provide security researchers with the necessary contact information and any additional details they might need to complete a vulnerability report.

Most of the companies surveyed converge on a single uniform format by putting the important contact information within a bulk of text.

Apple for example writes: 'to report security or privacy issues that affect Apple products or web servers, please contact: product-security@apple.com. You can use Apple's Product Security PGP key to encrypt sensitive information sent via e-mail.'[247] And Google states that 'if you believe you have discovered a vulnerability in a Google product, or have a security incident to report go to goo.gl/vulnz to include it in our Vulnerability Rewards Program. For Chrome vulnerabilities [...].'[248] Cisco approached the issue very differently by putting the necessary contact information into a simple table while also mentioning the hours in operation.[249] Additional information such as day's off, holidays, and a telephone number for emergency support can further contribute to enhance communication transparency.

Template:

Report a vulnerability	
E-mail	xxx@xxx.xxx (e-mails are acknowledged within xx hours)
OpenPGP Key	XXXXXXXXX
Hours	xx hours a day, x days a week, holidays
Phone Number (emergency contact)	+x xxxxxxxxxx

Note: Sending a confirmation of receipt email is essential to ensure that the security researcher knowns that his/her report has been received by your company. You can choose to either do this automatically or add a disclaimer clarifying your response time (see template).

E.1.3 Attributes of a good report

To receive the necessary information that will help verify and reproduce a security vulnerability in one of your products, it is critical to clearly state what kind of information you require from the security researcher.

While there are various ways of receiving an incident report, whether it is through a simple email, an online form, or a word template, it is important not to restrain security researchers in their input methods. Online forms for example usually do not allow images or video files to be attached, and Word templates might complicate user input due to possible formatting difficulties. Some companies employ very restrictive guidelines, such as Symantec which does not accept email attachments.[250] Other companies, such as Facebook, even encourage users to attach their own video files.[251] In the end a simple e-mail seems to be the best solution to maximise user flexibility while avoiding any loss of critical information.

---

247 Apple. n.d. 'Apple Product Security.' As of 15 September 2015: https://www.apple.com/support/security/
248 Google. n.d. 'Google Application Security.' As of 14 September 2015: https://www.google.com/about/appsecurity/
249 Cisco. n.d. 'Security Vulnerability Policy.' As of 14 September 2015: http://www.cisco.com/web/about/security/psirt/security_vulnerability_policy.html
250 Symantec. n.d. 'Vulnerability Management Commitment and Disclosure Policy.' As of 14 September 2015: https://www.symantec.com/security
251 Facebook. 2015. 'Whitehat.' As of 14 September 2015: https://www.facebook.com/whitehat

Template:

Attributes of a good report Quality before Quantity Please provide us with a detailed description and a clear and concise step-by-step guide in English to allow for the reproduction of the security vulnerability. (include screenshots where necessary) The step-by-step guide should include: - xxxxx - yyyyy - zzzzzz

E.1.4 Ineligible reports

Please make clear which reports and research methods your company will **not** accept or permit. For example, reports on login issues, password problems, spam, and suspected fraud activities ought to be handled by your support help desk.

Acceptable research methods should also explicitly differentiate between vulnerability scans and ethical hacks. While the difference is in some cases a grey area, a vulnerability disclosure policy should generally not enable researchers to hack into your company's systems. As such, DDoS attacks, brute-force attacks, malware installation, or making any kind of changes to your system (i.e. copying, deleting, and altering files) ought to be explicitly forbidden.

Bugcrowd and Facebook provide an exhaustive list of items they are not willing to accept as a security vulnerability,[252] while Lenovo states that 'all content other than specific security vulnerabilities in our products or services will be dropped.'[253] The Dutch National Cyber Security Centre (NCSC) specifically lists malicious acts that reporters must avoid to be free from any legal prosecution.[254]

Template:

Ineligible reports - Login issues and password problems - Spelling mistakes - HTTP 404 pages - Spam or suspected fraud activities - … **Not permitted acts:** - DDoS attacks - Brute-Force attacks

[252] Bugcrowd. 2015. 'Standard Disclosure Terms.' As of 15 September 2015: https://bugcrowd.com/resources/standard-disclosure-terms ; Facebook. 2015. 'Whitehat.' As of 14 September 2015: https://www.facebook.com/whitehat
[253] Lenovo. 2015. 'Lenovo Vulnerability Disclosure Policy.' As of 14 September 2015: https://support.lenovo.com/us/en/documents/ht103338
[254] NCSC. n.d. 'Responsible Disclosure.' As of 14 September 2015: https://www.ncsc.nl/english/securityt

- Social engineering
- Malware installation
- Making any changes to our system (incl. copying, changing, and deleting data)
- Sharing access with others

…

The xxx@xxx.xxx email is intended for the sole purpose of reporting a security vulnerability. If you are in need of technical assistance please contact our support help desk: yyy@yyy.yyy.

E.1.5 Procedural steps and timeline

Depending on a company's resources and expertise devoted to fixing security vulnerabilities, the layout of procedural steps and the self-imposition of a timeline widely differs.

Mature companies such as Google and Yahoo are committing themselves to publicly disclose vulnerabilities within 90 days,[255] while CERT/CC adheres to a 45-day deadline 'regardless of the existence or availability of patches or workarounds from affected vendors.'[256]

Companies that are new to the practice of security vulnerability disclosure are advised to **NOT** impose a 45- or 90-day timeline upon themselves. Instead the communication ought to focus on ensuring mutual respect (between your company and the security researcher), transparency, and greater flexibility.

Rapid7, for example, provides a series of steps they are committed to take to address the security vulnerability while also noting that they 'will endeavour to keep the reporter apprised of every step in the process as each step occurs.'[257] Symantec on the other hand clearly states that they 'cannot provide software security patches according to a set timeline. Each issue requires investigation, resolution, localization, and testing appropriate to its complexity.'[258]

Template:

Procedural steps and timeline

- Once we receive your vulnerability report, we will take every necessary step to investigate and resolve the security issue at hand in a swift and transparent manner.
- While we cannot provide patches according to a fixed timeline, we are committed to keep you informed at every step of the process.
- We request that you keep all communications regarding the vulnerability confidential, to ensure mutual trust and the flexibility to work with us towards the release of a patch, while guaranteeing an adequate timeframe for our customers to deploy said patch.
- We will publicly announce the vulnerability in our release note of the update and will mention the person/people who reported the vulnerability unless the researcher(s) wishes to remain anonymous.

[255] Yahoo. 2014. 'Users First: Our Vulnerability Disclosure Policy.' As of 14 September 2015: http://yahoopolicy.tumblr.com/post/104777538533/users-first-our-vulnerability-disclosure-policy; Google. n.d. 'Google Application Security.' As of 14 September 2015: https://www.google.com/about/appsecurity/
[256] CERT/CC. n.d. 'Vulnerability Disclosure Policy.' As of 15 September 2015: http://www.cert.org/vulnerability-analysis/vul-disclosure.cfm?
[257] Rapid7. n.d. 'Vulnerability Disclosure Policy.' As of 15 September 2015: https://www.rapid7.com/disclosure.jsp
[258] Symantec. n.d. 'Vulnerability Management Commitment and Disclosure Policy.' As of 14 September 2015: https://www.symantec.com/security

ENISA
European Union Agency for Network
and Information Security
Science and Technology Park of Crete (ITE)
Vassilika Vouton, 700 13, Heraklion, Greece

Athens Office
1 Vass. Sofias & Meg. Alexandrou
Marousi 151 24, Athens, Greece

Catalogue Number TP-01-15-893-EN-N

PO Box 1309, 710 01 Heraklion, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu

ISBN: 978-92-9204-148-9
DOI: 10.2824/610384
Cat.Number:TP-01-15-893-EN-N