Enhancing ISP-Consumer Security Notifications

by

Nathaniel H. Fruchter

B.S. Decision Science
B.S. Science, Technology, and Public Policy
Carnegie Mellon University (2016)

Submitted to the Institute for Data, Systems, and Society
in partial fulfillment of the requirements for the degree of

Master of Science in Technology and Policy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2019

Author_____

Technology and Policy Program
January 18, 2019

Certified by_____

Dr. David D. Clark
Senior Research Scientist, MIT CSAIL
Thesis Supervisor

Accepted by_____

Dr. Noelle Selin
Director, Technology and Policy Program
Associate Professor, Institute for Data, Systems, and Society
and Earth, Atmospheric and Planetary Science

Enhancing ISP-Consumer Security Notifications

by

Nathaniel H. Fruchter

Submitted to the Institute for Data, Systems, and Society
on January 18, 2019, in partial fulfillment of the
requirements for the degree of
Master of Science in Technology and Policy

Abstract

Security notification schemes hold great promise for improving both consumer cybersecurity and general network health as malware and other sources of malicious activity are becoming more prevalent on home networks. For example, botnets of Internet of Things devices engage in denial of service (DoS) attacks and ransomware holds data on personal and commercial systems hostage. Many of these threats are relatively opaque for an end user. An end user may not know that their smart device is participating in a DoS attack at all, unless they notice a protracted slowdown in network speeds.

An upstream network provider like a consumer ISP has more visibility into the issue. Due to their privileged position, ISPs often have more data about the status of a malware infection, denial of service attack, or other malicious activity. This extra information can be of great benefit for the purposes of notification. For instance, an ISP may be able to *notify* a customer that a device on their network is being used for a DoS attackor that they see communication with a server involved in distributing ransomware.

ISPs and other organizations that try and implement these schemes often run into a set of questions: *How do I get the right data to power the notification? How do I ensure the user trusts the notification? Can I ensure the notification is not spoofed? Is there an optimal way to present the notification? How do I make sure a user takes the proper remedial action?*

This thesis presents a framework for new notification schemes to answer these questions by examining four key elements of a notification: *form, delivery,* and *content.* It also proposes *multi-factor verification*, a novel scheme to address trust and spoofing issues within a notification scheme. Finally, it provides a model for a new ISP-user security notification scheme within the context of the United States market and policy landscape.

Thesis Supervisor: Dr. David D. Clark
Title: Senior Research Scientist, MIT CSAIL

Acknowledgments

I've had the privilege of knowing and working with many great people during my short stay at MIT.

To my advisor, Dr. David Clark, and Steve Bauer: thank you both for your guidance, mentorship, and trust. This thesis has changed and evolved from the early discussions we had when I joined the lab, but your support has been a constant. Your encouragement to explore new projects and ideas was invaluable

To Cecilia, Leilani, Sam, Jonathan, Grace, Ilaria, Mike, George, Philipp, and my other IPRI lab mates (both current and former): thank you for the support, laughs, and numerous coffee runs. More importantly, thanks for being such great friends and colleagues. 

To Danny, Taylor, Karen, Bill, Arthur, Sue, Amy, Mel and the rest of ANA and IPRI: thank you for your help, support, and leadership. Thanks for making many interesting conversations and opportunities happen.

To my TPP friends and classmates: I've had the pleasure of knowing a lot of you (TPP'17, 18, 19, and 20!) and you're all the absolute best. Thank you for your friendship, support, and camaraderie. Keep doing great things!

To Barb, Frank, Noelle, and the TPP and IDSS staff: thank you for helping to make TPP a second home for all of us here at MIT. We wouldn't be here without you!

This thesis also would not have happened without some inspiration from a 2017 internship at the National Telecommunications and Information Administration (NTIA), part of the U.S. Department of Commerce. *To Allan Friedman and Edward Carlson*: thank you for providing a great internship experience and an excellent introduction to the D.C. Internet policy community.

Finally, as always, my family has been a source of constant love and support. Thank you for everything.

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

List of Sidebars

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

Introduction

For most people, October 21, 2016 was not an especially notable date. If you were traveling through London, you may have noticed the evacuation of London City Airportor, perhaps, you heard about a medium-sized earthquake in southern Japan [50, 58]. However, if you were a system administrator, network operator, or on the eastern seaboard of the United States, 10/21/2016 may be a bit more memorable. The 21st marked the beginning of the Mirai botnet's cyberattack against Dyn Inc, along with the associated ripple effects that would lead to service degradation and outages for many.

If you were a consumer or end user, the first effect you would have noticed would be the swath of slow or inaccessible websites. Much of the day's media attention focuses on this aspect. For example, WIRED discusses the day's "massive east coast Internet outage", the Wall Street Journal notes that "dozens of popular websites [were] unreachable for part of the day", and the BBC reports how "Reddit, Twitter, Etsy, Github, SoundCloud, Spotify and many others were all reported as being hard to reach." These availability and quality of service issues may have been problematic or annoying, but they were temporary: the hardest-hit sites became available by the end of the day and any consumer-facing disruptions seemed to be temporary.

For system administrators and network operators, though, the day's cyberattacks did not bode well. The initial October 21 outages occurred because Mirai targeted the servers of Dyn Inc., a managed network services provider that hosted the primary domain name system (DNS) servers for many popular websites. While Internet infrastructure providers are often subject to attacks, Mirai's ability to create over 600 gigabits per second of traffic created a distributed denial of service

(DDoS) of unprecedented scale and volume [55]. How was this possible? By leveraging insecure Internet of Things (IoT) devices, the botnet's authors were able to create an effective, high volume, and decentralized source of traffic which they could target practically on demand (see Sidebar 1 for more details).

Sidebar 1 - Dissecting the Mirai botnet

What is a botnet?

A botnet is a term used within the computer security community to refer to a network of computers compromised by an adversary or malicious actor [15]. An individual machine in the network is referred to as a bot. Bot activity is orchestrated through command and control (often abbreviated C2 or C&C) protocols by a botmaster. Botnets have long been a cybersecurity threat; the ability to co-opt and aggregate others' computing and network resources is obviously attractive.

What is Mirai?

Mirai is the name given to a family of botnets derived from a common codebase. It differs from most prior botnets. It targeted insecure embedded and Internet of Things (IoT) devices like streaming cameras, routers, and printers. Many of these devices were shipped from the factory with poor or nonexistent security, including weak default passwords for root accounts. The authors of Mirai took advantage of these insecure defaults. Armed with a list of known credentials, bots would propagate themselves scanning a large swath of the IPv4 address space in a pseudorandom fashion, logging in with credentials, and using these scan results to create a "candidate" list of new bots. At a later date, a "loader" program would install a malicious binary onto a candidate and add the newly-infected bot to the network.

How big was Mirai?

Antonakakis et al. [2] study provide the first deep analysis of Mirai and give a rough estimate of 1.78 million potential members of the botnet over their observation period. Of those, they are able to positively identify 587,743 active devices and estimate a "brief peak of 600,000 devices."

How did Mirai cause so much trouble?

Members of the Mirai botnet engaged in a distributed denial of service attack, or DDoS. As the botnet had a fairly large, distributed footprint of IoT devices, the botmaster was able to direct a large volume of traffic towards a particular DDoS target (such as Dyn). IoT devices often have access to limited bandwidth and computational capacity, so the botnet's size was crucial to this attack's success.

> Researchers speculate that based on observed DDoS targets, Mirai's botmaster(s) were targeting gaming services like Sony's PlayStation Network and Microsoft's Xbox Live. In fact, analysis demonstrates that initial Dyn-hosted targets were actually DNS servers for the PlayStation Network. This indicates that any spillover to the rest of Dyn's clients was accidental. However, since Dyn was a key DNS provider for many large Internet services, the DDoS attack was able to prevent the successful resolution of IP addresses and deny service to an uncommonly wide number of users.

1.1 Threat Data, Notification, and Remediation

Understandably, Mirai constituted a nightmare scenario for those trying to clean up the attack's aftermath and head off future waves of disruption. Let us take a hypothetical incident response team at a consumer Internet service provider (ISP) affected by the botnet. As news and data about the scope of the attack begin to pour in, many questions present themselves to the team. These questions might include:

- Where are the bots located?
- How can the team find out about new infections on the ISP's network?
- How can affected customers be warned that they're participating in a botnet?
- How can the team remove devices from the network of infected bots?
- How can uninfected devices on the ISP's network be prevented from joining the botnet?

These questions all revolve around the concepts of **threat data**, **notification**, and **remediation**– concepts that are key to addressing any sort of large cybersecurity incident at ISP scale. If an ISP is able to leverage *data* about cybersecurity threats, *notify* affected parties, and have them *remediate* the issue, we can claim that the ISP is able to deal with the threat in an effective manner.

1.1.1 Example

What might this look like? Let's address each briefly within the context of our team responding to the Mirai attack.

The concept of **threat data** addresses the first two questions in the list: *Where are the bots located? How can the team find out about new infections on the ISP's network?* Put another way, our ISP's response team may know that an attack is occurring, but may not know much else. Before they can begin addressing the questions of bots resident on their own network, they will have to find a reliable way to detect or fingerprint infected devices. They will also have to rely on internal and external

sources of threat data to assemble a complete picture of the threat on their network.[1]

After the incident response team has a better idea of how Mirai has affected the ISP's network, they next turn to the question of **notification**: *How can affected customers be warned that they're participating in a botnet?* This could take any number of forms, from a letter to an email, popup, or push notification.[2] Notification is especially crucial in the case of Mirai as its constituent bots are embedded and IoT devices, often left unattended and unmaintained by their owners and users.

1.1.2 Remediation

In most cases, notification isn't too useful unless it's paired with **remediation**, or a set of follow-up action(s) that will fix or correct the problem at hand. In the case of Mirai, remediation centers on two questions: *How can the team remove devices from the network of infected bots? How can uninfected devices on the ISP's network be prevented from joining the botnet?* In hindsight, we now know that Mirai did not persist across devices reboots or firmware resets [2]; this means that advising notified users to reboot or reset their devices would have been a useful remediation strategy. In the absence of this knowledge, though, remediation could constitute anything from an automatic action on the ISP end (e.g., firewalling or quarantining devices) to a manual action on the user's end (e.g., applying a firmware patch).

1.2 Why Didn't This Happen?

With our three key concepts in mind–threat data, notification, and remediation–we can now see how an ideal response to Mirai could have taken shape. By leveraging internal security data and external threat data from public and private partners, the ISP could have quickly assessed the extent to which Mirai was resident on its network. This data could have also identified specific devices or address blocks that were part of the botnet. The incident response team could then feed this data into a notification process. Through automatic and manual processes, the ISP could inform affected customers about the infected status of their devices using various notification channels. Finally, with knowledge of Mirai's weaknesses, the ISP could offer potential remedial actions alongside the notification.

The power of this ideal *notification and remediation* ("N&R") flow can't be understated. If ISPs were able to coordinate this type of response to Mirai, they could largely head off the botnet's

[1] These may include internal monitoring systems, data sharing agreements with other ISPs and industry partners, and data sourced from agreements with governments and other public sector stakeholders.

[2] Additional options are expanded on in chapters 2 and 3.

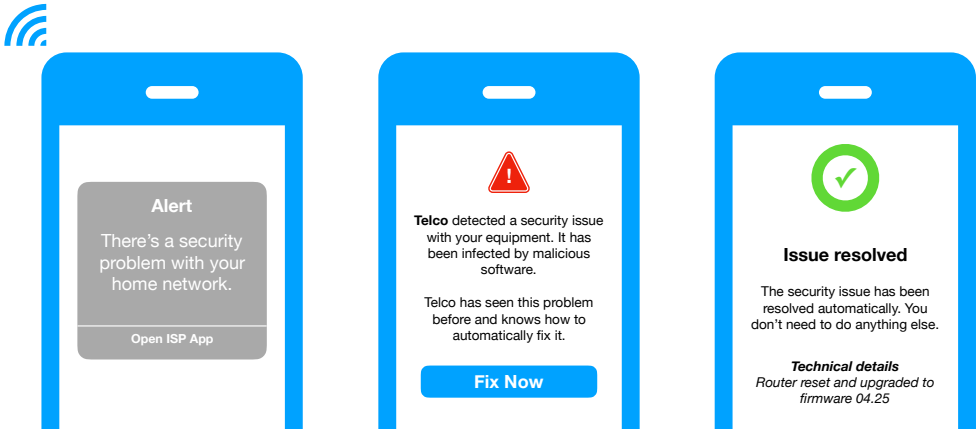

Figure 1-1: Sketch of an idealized notification and remediation process

negative impacts. Instead of sustaining a peak size of almost 600,000 bots, effective N&R could have prevented Mirai from gaining a foothold in vulnerable IoT devices. Even if devices were re-infected after a period of time, thinning Mirai's ranks would almost certainly have a net positive of the health of the Internet ecosystem. Dyn may have stayed down for a shorter period of time, fewer Internet users would have seen quality of service impacts, and network operators would have a better idea of the scale of the remaining problem.

However, this isn't what happened in the aftermath of Mirai. Recovery from the cyberattack took on the order of days, not hours. Websites, such as the one run by security researcher Brian Krebs, were brought back online through the use of "DDoS shields" [38] that had enough excess bandwidth to absorb the brunt of the attack. Heroic technical and law enforcement efforts stemmed the attack down to manageable levels, but comparatively little was done on the notification and remediation front [2]. Years later, security researchers still see variants of Mirai floating around on the Internet.

1.3 Looking Ahead

A central question remains: why couldn't ISPs and other network operators utilize notification and remediation effectively in the face of Mirai? Why is the user experience sketched in Figure 1-1 still a *sketch*, not a reality? This thesis is motivated by that question. Put another way: with notification and remediation showing so much promise, why is its implementation and use less than optimal?

The rest of this thesis addresses that question by tackling it from several angles. First, it examines the history of the notification paradigm. Chapter 2 overviews prior attempts at notification and remediation systems, identifying failures and successes from a rich body of prior work. It also addresses the question of sharing threat data. Next, Chapter 3 discusses several notable challenges facing good notification and remediation systems (e.g., changing home network topologies and the difficulty of warning design). It does so through the lens of analogous challenges in related fields of work.

After establishing this background and context, the thesis then turns to the design of a *feasible*, *trusted*, and *usable* notification system. Chapter 4 lays out guidelines for *constructing* a modern notification system, taking into account previously identified design and security challenges. Chapter 5 lays out recommendations for *implementing* a model notification system and takes into account organizational and policy considerations while doing so. Finally, Chapter 6 lays out some key conclusions and avenues for future work while Appendix 1 sketches out a multi-part user study to realize one avenue for future work.

1.3.1 Contributions

This thesis builds on prior work in the consumer notification and remediation space. Specifically, it:

- Provides a comprehensive overview of prior notification and remediation efforts
- Identifies challenges and pain points in current notification and remediation processes
- Integrates and applies research from the behavioral science and human-computer interaction communities to the notification and remediation context
- Provides a design framework for an improved notification system
- Lays out a threat framework and identifies novel solutions for security and verification problems in the notification space

- Gives actionable implementation suggestions for ISPs with existing notification systems through a generalized case study

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 2

Initiatives and Standards

In May 2017, U.S. President Donald Trump issued Executive Order 13800 [49]–often dubbed the "cybersecurity executive order". Among other actions, it directed the Departments of Commerce and Homeland Security to "to encourage collaboration with the goal of dramatically reducing threats perpetrated by automated and distributed attacks (e.g., botnets)". Of course, discussion of remediation techniques figured into the resulting report [20]. Notably, though, the report also makes a passing reference to many "previous efforts" in the area. This hints at the surprisingly large and diverse body of work on in the area. Numerous schemes, initiatives, and solutions have been proposed to deal with threat data, notification, and remediation–the exact problem areas introduced in this thesis.

So, what previous efforts exist in the U.S. and elsewhere? More importantly, why haven't they gained much traction? And why haven't they figured into our current discussions on notification and remediation? This chapter discusses the historical origins of notification and remediation in various Internet security contexts. It also provides an overview of current and prior N&R initiatives and concludes with a discussion of new N&R paradigms.

2.1 Historical origins

Notification and remediation aren't new concepts. However, contemporary treatment of the N&R lifecycle as a unit including the ISP likely began in the late 1990s and early 2000s with the emergence of email spam and PC botnet threats. At this point in time, many stakeholders–ranging from

national governments to ISPs, network operators, and manufacturers–struggled with ways to detect, prevent, and mitigate botnets. This struggle was intertwined with the fight against email spam, often generated by those same botnets.

Stakeholders soon realized that collaboration was necessary. The need for collective action against these new, distributed threats was paramount; individuals or individual organizations found themselves in need of resources, expertise, and cooperation from other stakeholders. The Internet community has a long history of multi-stakeholderism and self-regulation, so this need led to both formal and informal cooperative structures.

Common modes of cooperation included the creation of regional efforts (often involving governments), the creation of multi-stakeholder industry bodies, and the creation of standards and best practices. For example, Australia's Communications and Media Authority convened a successful anti-spam working group [43] and stakeholders from the public and private sectors collaborated through new organizations like the Messaging, Malware and Mobile Anti-Abuse Working Group (M3AAWG) and Spamhaus [16].

These groups took on a wide variety of activities. Groups like Spamhaus created and distributed spam blocklists and blacklists used by security software, email providers, and network operators. Public-private partnerships found themselves advocating for legislative change, often found through laws like the U.S.'s CAN-SPAM Act of 2003 and Australia's 2003 Spam Act.[1] Partnerships like M3AAWG, along with stakeholders at various Internet standards bodies like the IETF, also made progress on the technical front.

While these organizations and partnerships did not deal directly with notification as it's construed in this thesis, they did something equally important and laid the groundwork for collaboration against online threats. As such, these efforts found themselves largely successful, able to make a significant dent in that era's spam and botnet problem [16].

2.2 Existing initiatives

However, the threats at hand began to evolve. Stakeholders responded by both broadening the scope of these existing anti-spam and anti-botnet collaborations and creating new initiatives. These initiatives turned into the first coordinated notification and remediation initiatives.

This section summarizes and categorizes "prior art" in the N&R world by cataloging public and private N&R initiatives. While there are many Internet and cybersecurity initiatives out in the wild,

[1] Found at *15 USC §103* and Aus. Statute *Spam Act 2003 (Cth)*, respectively.

we limit ourselves to those that are *primarily concerned with end-user notification and remediation.* We categorize these initiatives around four core activities: threat data sharing, end-user notification, remediation assistance, and end-user education. Efforts usually engage in multiple activities, but not all do.

A typical **threat sharing hub** takes in data from various sources including ISPs, computer emergency response teams (CERTs), and law enforcement, usually consisting of infection reports tied to IP addresses and ports. Data is then distributed to ISPs and network operators for further action. These hubs may also share best practices.

User notification initiatives usually consists of contact from an ISP or other notifying party. These notifications inform users that malicious activity has been detected from their machine(s) or accounts. The method of contact varies significantly and can include phone, email, postal mail, web page overlays, or a more drastic walled garden approach which temporarily restricts internet usage.

Approaches to **remediation assistance** are also diverse. Strategies include generic advice pages, requests to contact the ISPs customer support, and provision of anti-virus or disinfection tools by the ISP.

Finally, **end-user education** usually takes the form of generic or tailored security advice (similar to remediation assistance) and communication of best practices. Approaches to education may also include public messaging and awareness campaigns. Certain efforts have also engaged stakeholders through presentations and community workshops.

2.2.1 Australian initiatives

Australian Internet Security Initiative: regional threat sharing hub, notification assistance.

The Australian Internet Security Initiative (AISI) is a data-sharing system currently administered by AusCERT, the country's computer emergency response team. The Initiative is a public-private partnership and acts as a clearinghouse for data.

The AISI originated as part of the Australian Communication and Media Authority's anti-spam mandate, but quickly grew into a general purpose hub for sharing threat data [42]. ISPs claim ownership over blocks of IP addresses and subscribe to automated email notifications about threats detected from those address blocks. Data is shared in plain-text, comma-delimited format. The

AISI also provides a summary dashboard for subscribers [66]. This data can then be leveraged by ISPs to send notifications.

The AISI seems to be highly effective due to its position. An in-depth evaluation of the program [66] discusses its failures and successes with small, medium, and large ISPs in the Australian market. The report notes that many ISPs "solely" rely on the AISI for malware reports and that data shared is seen as useful by ISP recipients. The report also briefly discusses the perception of notifications by users. The credibility of an AISI report as government-sourced is noted as "[enhancing] the notification's legitimacy", leading to "very few" unresponsive recipients after notification.

The AISI is closely related to **iCODE**, a "voluntary code of practice for industry self-regulation in the area of cybersecurity" put forth by Australian ISPs. The code suggests ways that ISPs can help educate and protect their customers and recommends use of the AISI [19].

2.2.2 European initiatives

Advanced Cyber Defence Centre: Regional threat sharing, remediation, research, and educational partnership.

The Advanced Cyber Defence Centre (ACDC) was a 30 month pilot that ran from 2013 to mid-2015 that partnered with academic, industry, and government stakeholders. Through these partnerships, it ran technical experiments, provided support for 8 regional support centers, and created a "data sharing hub" for the European market.

The ACDC's regional support centers were run by local partners (including ISPs and local security organizations). The centers were generally branded with the "Botfree" or "Antibot" name, but this largely varied by country.[2] Most provided general advice and educational material along with an ACDC-developed "EU-Cleaner" antivirus tool. Some, such as the Spanish National Cybersecurity Center's *Servicio AntiBotnet* went further, providing browser extensions that also handled security notifications [56]. While the ACDC's mandate has expired, many of the regional support centers are still active and supported by their local sponsors.

Finally, the ACDC seems to have prototyped its threat sharing hub [70], but it does not appear to have left the prototype stage.

Autoreporter: Finnish threat data sharing network

Autoreporter was a Finnish threat data sharing hub created by CERT-FI, the country's computer

[2] See https://acdc-project.eu/8-support-centres/ for a list of the active centers.

emergency response team. It helped CERT-FI automate the sharing of trusted threat data to Finnish network administrators [33]. Sharing was done in a similar manner to the AISI (and SISSDEN, Autoreporter's contemporary): ISPs and network operators would receive machine-readable data via email. Autoreporter has since been rolled into the AbuseHelper open source project.[3]

AbuseHub: Dutch threat data sharing network

AbuseHub is a national threat data sharing hub that specifically targets botnet mitigation in the Netherlands. Like its comparable projects, AbuseHub ingests a wide variety of threat data and forwards it on to ISPs and other owners of address blocks. An evaluation of AbuseHub notes its wide reach–over 90% of Dutch ISPs subscribed to the service's data [23]. The same evaluation finds that the service "unequivocally" improved the scope and quality of threat data received by ISPs, but also concludes that more work is needed to scale up botnet cleanup, create best practices for notification and remediation, and incentivize more sharing of data.

SISSDEN: Regional threat data sharing network

Although the ACDC's threat sharing hub has not progressed, the European Union has funded another similar project. The Secure Information Sharing Sensor Delivery Event Network (SISSDEN) is a regional data sharing network whose primary objective is "to offer National CERTs, ISPs and network owners free reports on malicious activity detected on their networks" [61]. This goal is similar to that of Australia's AISI; the network's sole focus is on gathering and disseminating threat data.

SISSDEN currently lacks evaluation data as it began development in 2016 and is scheduled to end in April 2019.

2.2.3 Japanese initiatives

Cyber Clean Center: regional notification and remediation, research, and botnet takedown.

The Cyber Clean Center (CCC) is a Japanese public-private partnership between ISPs, anti-virus companies, and the Ministry of Communications. It aims to address the proliferation of botnets on Japanese networks. It was established in 2006 "for the purpose of reducing the number of botnet-infected computers to as close as zero as possible" and was chartered to last 5 years [17]. Its work had three main emphases: botnet takedowns, malware research, and infection prevention.

[3] A "framework for receiving and redistributing abuse feeds" which sees sporadic development activity on GitHub at https://github.com/abusesa/abusehelper.

The CCC leveraged its unique partnerships to create tailored remediation tools for specific botnet threats. Users were identified by ISPs, notified through phone, email, and regular mail, and provided with relevant remediation tools. The CCC also tracked incidents tied to users through unique host IDs which allowed tailored follow up notifications for incidents and re-infections. An evaluation [17] deemed the CCC a success, reducing overall infection rates even as the number of broadband users in the country grew. Similar to the AISI, the report also noted how credibility of warnings was enhanced by coming from a validated Japanese government source.

2.2.4 South Korean initiatives

Cyber Curing Center: regional remediation assistance and education

South Korea created a country-specific remediation call center dubbed the Cyber Curing Service. It is also known as the 118 Call Center after its phone number. The Cyber Curing Service was originally targeted towards cybersecurity advice about botnet threats, but has evolved since. A 2015 report by the Korean Internet Security Agency notes the creation of a "Nationwide 118 Information Security Support System" that aims to provide education, advice (remediation assistance), and "emergency response" capabilities for small and medium-sized businesses [37].

2.2.5 United States initiatives

Notification and remediation initiatives in the United States have mainly been internal to ISPs. However, existing anti-botnet and anti-spam collaborations still participate in the discussion.

ABC for ISPs: notification and remediation code of conduct

The Anti-Bot Code for ISPs (ABC for ISPs) is an industry code of conduct for U.S. ISPs. It was created out of work from M3AAWG and a Communications Security, Reliability and Interoperability Council (CSRIC) chartered by the U.S. Federal Communications Commission and was finalized in 2012 [27, 28]. The ABC recommends notification and remediation as a primary strategy to reduce botnets in the U.S. Internet ecosystem. Most major ISPs in the U.S. subscribe to the code, including AT&T, Comcast, CenturyLink, Cox, and Spectrum.

Multi-stakeholder consultations

The U.S. government has also initiated two requests for comment on the topic of notification and remediation. A **2011 Request for Information** asks about "requirements of, and possible approaches to creating, a voluntary industry code of conduct to address the detection, notification

and mitigation of botnets" [44]. This RFI involved many of the same stakeholders as the ABC for ISPs, but was initiated by the Departments of Commerce and Homeland Security.

Six years later, the Departments of Commerce and Homeland Security issued another Request for Information. This **2017 RFI** was on "botnets and other distributed threats" as specified by Executive Order 13800 (see the beginning of this chapter). The RFI resulted in a 2018 report [20] which presented 5 high-level goals for improving the resilience of Internet infrastructure. While not at the forefront, ISP-centric notification and remediation are discussed alongside suggestions about data sharing, adaptation to threats, and better endpoint mitigation techniques.

2.2.6 IETF activity

The Internet Engineering Task Force (IETF) supports the publication of Requests for Comment (RFCs). RFCs are often used to communicate Internet standards, but are also used as informational publications by the Internet community. Two RFCs on notification and remediation have been published by staff from Comcast, a large U.S. cable ISP.

RFC 6108 *(ca. 2011)* describes "Comcast's Web Notification System Design" [13] which provides "critical end-user notifications to web browsers." These notifications include security notifications about traffic "patterns that are indicative of malware or virus infection." The RFC provides the technical design of such a system and also discusses considerations for deployment, security, and the system's merits over other alternatives.

RFC 6561 [40] *(ca. 2012)* provides "Recommendations for the Remediation of Bots in ISP Networks" and primarily concerns itself with possibilities for bot detection, notification, and remediation. The bulk of the RFC describes various methods of notification (ranging from physical mail to SMS and email) and remediation (including guided or "professionally-assisted" processes). It also briefly touches on operational considerations such as user-to-ISP feedback and what to do if a user refuses to remediate.

2.3 Network-edge notifications

What if your router was smart enough to create its own notifications? Outside of the broad-based initiatives discussed in this chapter, the promise of N&R has also gained the attention of vendors in the "smart home" ecosystem. This means that developers, manufacturers, and hardware vendors are beginning to give us their take on the N&R paradigm. We briefly discuss this emerging area of work here.

Two emerging categories of home network device are the *Internet of Things gateways* and the *smart router*. Smart home hubs are essentially routers for IoT protocols and centralize communication with sensors and IoT devices from various manufacturers. Smart routers blend traditional home network routers with cloud administration tools and managed security services.

Can a smart device at the edge of a network see enough to notify users about security issues? Some manufacturers think so. The Norton Core [48] and F-Secure Sense [46] both leverage their vendors' security expertise. Administered through smartphone apps, these routers notify users about security issues gleaned from local *(e.g., deep packet inspection and port scanning)* and external *(e.g., fingerprints, blacklists)* sources of data. They also allow for some types of remediation–the Norton Core allows users to automatically firewall suspicious devices, for instance.

This approach seems promising, but there are still hurdles to overcome. Routers can only do so much on the remediation angle and a reliance on manufacturers for threat data lends these devices to a subscription model. The promise of smart devices at the network edge should not be discounted, though. If network-edge notifications by routers and hubs can be integrated into the larger N&R ecosystem–and if they can work in concert with other schemes at the ISP level–these devices can provide a complementary service to the ecosystem. In concert with other N&R techniques, they can provide an excellent first or last line of defense, as well as another root of trust in the ecosystem.

		Leadership		Services				
Scope	**Initiative**	**Government**	**Industry**	**Threat Sharing**	**Notification**	**Remediation**	**Education**	**Active?**
Australia	**iCODE**	originated	administers	✓	—	✓ (generic advice)	Mentioned	✓
Australia	**Australian Internet Security Initiative**	originated	participates	✓	in partnership w/ISPs	✓ (ISPs)	—	✓
EU	**Advanced Cyber Defence Centre**	research grant	administers, in partnership	✓	some countries	✓ (generic advice)	ACDC; ISP	
EU	**SISSDEN**	research grant	participates	✓	—	—	Mentioned	✓
Japan	**Cyber Clean Center**	originated; coordinates	Provides expertise; receives info	✓	in partnership w/ISPs	✓ (tailored advice)	yes; govt through ISP	—
Netherlands	**AbuseHub**	research grant	Participates; provides + receives data	✓	—	—	n/a	*successor*
South Korea	**Cyber Curing Service**	originated; coordinates	Stakeholders, provides tools and expertise	✓ *(from gov't)*	—	✓ (guidance)	ISP and govt	—
South Korea	**118 Call Center**	Administered	Unknown	—	—	✓ (hotline)	gov't	?
USA	**Anti Botnet Code for ISPs**	coordinated through FCC	Expertise; stakeholders	Mentioned	Mentioned	Mentioned	Mentioned	✓
USA	**2011 RFI**	Issued RFI	Comment as stakeholders	Mentioned	Mentioned	Mentioned	Mentioned	—
USA	**2017 RFI**	Issued RFI	Comment as stakeholders	Mentioned	Mentioned	Mentioned	Mentioned	—
—	**RFC 6108**		Authored as stakeholder	—	Mentioned	Mentioned	—	✓
—	**RFC 6561**		Authored as stakeholder	Mentioned	Mentioned	Mentioned	Mentioned	✓

Figure 2-1: Summary of notification and remediation initiatives

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 3

Why Better Notification Schemes Are Needed

The landscape of prior work in the notification space is undoubtedly rich. Now that we have an idea of what those initiatives looked like, how do they stack up in the face of today's modern challenges?

3.1 Issues with existing schemes

Here, we detail five common issues seen across existing notification schemes. These issues largely serve as roadblocks that impede these schemes' effectiveness in the face of modern threats, such as the Mirai botnet mentioned in Chapter 1.

3.1.1 Changes in Internet and device usage

The composition of home networks is changing at a rapid pace. Simple networks involving a router and one or two computers are a relative rarity today. More complex topologies that integrate "smart home" hubs, IoT devices, and multiple devices (laptops, tablets, smartphones) per user are quickly becoming the norm.

Unfortunately, most notification schemes assume that today's home networks look like their simpler, older counterparts. This is not the case. For example, Japan's Cyber Clean Center is the only

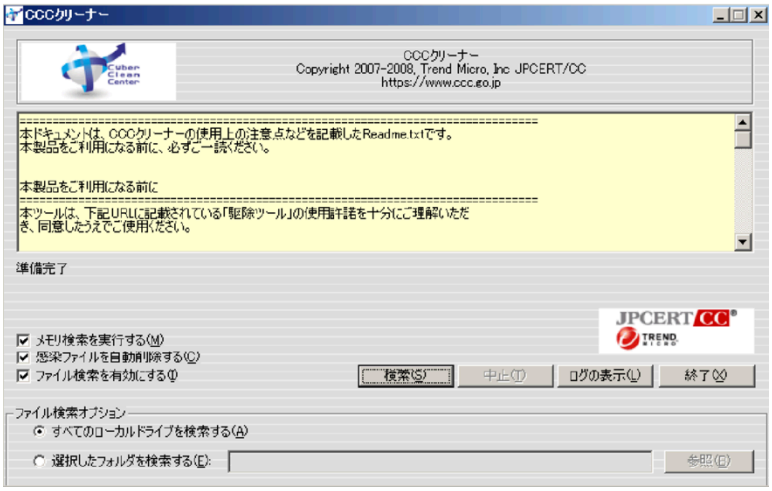

Figure 3-1: Japan Cyber Clean Center remediation tool

initiative we see that creates tailored remediation tools–obviously a good thing. However, the tools used by the CCC are targeted specifically towards machines running Windows (Figure 3-1). Similarly, recommendations from European ACDC support centers and U.S. ISPs both assume the presence of a desktop or laptop running Windows or macOS. Their generic remediation is unusable in the face of a security threat stemming from other platforms including IoT and mobile devices.

Even if remediation instructions were provided for IoT devices, we run into another problem: seeing the notifications themselves. RFC 6108 [13] describes a system that injects notifications into web pages through a transparent proxy system (Figure 3-2. This relies on the assumption that the notified party will be using a web browser to view HTML that can be rewritten. This is often not the case. For instance, dynamic web apps often use JSON or XML as the primary medium of data exchange. Many users also use tablets or smartphones as their primary devices and these platforms' apps usually exchange data through defined APIs instead of rendering HTML.

Finally, schemes don't account for changes in Internet security practices. For instance, many sites and services now use HTTPS to encrypt their traffic. In fact, the majority of traffic to Google is encrypted, with the company citing a 94% HTTPS share across its services.[1] In these cases, HTTP rewrite notifications [13] are impossible without breaking HTTPS–a huge breach of user trust and security.

[1] As of December 2018, see https://transparencyreport.google.com/https for updates.

3.1.2 Delivery and presentation

Many ISPs make an effort to have their customers interact with notifications. After all, emails can be deleted, phone calls can be ignored, and physical letters can be thrown away. However, current approaches to notification (besides Comcast's RFC 6108 and the Spanish *Servicio AntiBotnet*'s browser extension) use emails, phone calls, and letters as their primary means of notification. Existing warnings and communications also don't follow best practices for the communication of risk (see Section 4.2).

Clearly, more work is needed on creating well-designed notifications that fully leverage the capabilities of our networks and devices. This sentiment about necessary technical and design work is echoed by many reports, but is left as "future work" in most. The evaluation of AbuseHub, section 13 of RFC 6108, Action 3 of the Executive Order 13800 report, "next steps" in the AISI's evaluation, and the "Guide to Barriers to Code Participation" of the ABC for ISPs all mention the need for work on design and delivery [23, 13, 20, 66, 28].

3.1.3 Remediation advice

In addition to the overall presentation of the notification, we should also pay special attention to any remediation advice that is given. Why? Remediation is a difficult task! In the face of complex security issues, users often have trouble interpreting warnings and taking the appropriate action [8]. With this in mind, ISPs must strike a delicate balance when providing advice. Should they attempt to give advice tailored to specific customers or threats, or should they relay more generic security advice in the hopes that it will help the largest majority of customers?

A few examples demonstrate the power of tailored remediation advice. Japan's Cyber Clean Center successfully created tools for specific botnet threats, lowering the barrier to entry for remediation as users did not have to think about which strategy or tool to apply. The Korean Cyber Curing Center mentions a support hotline approach, but is scant on details. Finally, some ISPs, like the exemplars mentioned in the Australian Internet Security Initiative's evaluations, were able to give tailored remediation advice through conversations with ISP technical support [66].[2]

However, most schemes–especially those administered by ISPs–keep things broad. Figure 3-2 shows sample remediation advice from Comcast, a U.S. ISP. Users are given general advice about anti-malware tools and keeping software updated, but aren't made aware of any specific actions to be

[2]One common concern with this approach is its scalability. It appears that the AISI participants had relatively few customers that needed personal assistance, allowing for successful use of this strategy.

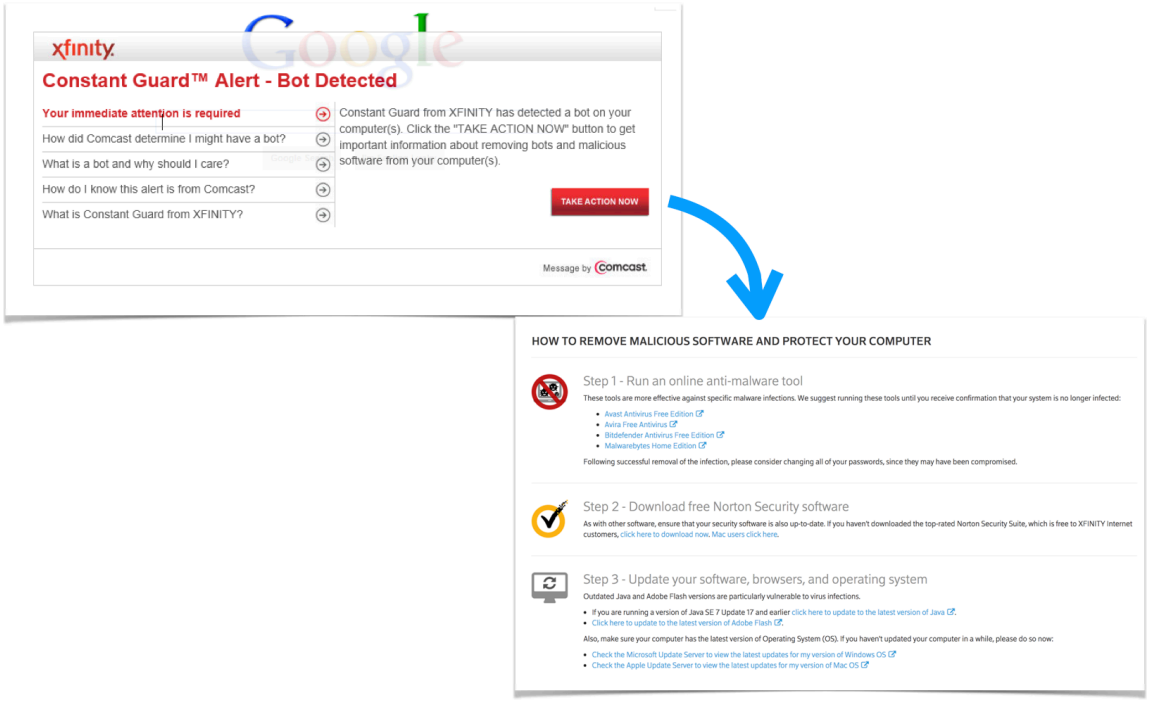

Figure 3-2: Example HTTP rewrite notification from a major consumer ISP.

taken in the face of *their* threat. The Botfree.eu campaign takes a similar tack and provides general advice on removing malware. Generic advice is not always ideal, especially for non-expert users. In the face of more complex threats on more complex networks, more tailored remediation advice will be needed–especially if there's a mismatch between the advice and threat.

Take our Mirai example: if an IoT camera is infected, users will be confused and unable to take action if they only know how to disinfect a PC.[3] Even if those users do take the correct action, they will also lack any substantive feedback about the remediation's success or failure. This combination of confusion and uncertainty does not make for a good user experience, especially in the security context.

3.1.4 Trust

Work is also needed on ways to improve trust in notifications. For instance, a Reddit user posting in a tech support thread notes their alarm at seeing an HTML rewrite notification from Spectrum, a large U.S. ISP [62]:

> *If this is legitimate, I don't know why the hell they don't contact me in a normal way like email or a phone call instead of hijacking a freakin' wordpress blog. Are they trying to look as suspicious as possible?*

These worries are compounded by a lack of public communication or documentation of these notifications. Spectrum makes no public documentation about these alerts available. This lack of information makes it even more difficult to verify the authenticity of the notification; users must resort to customer support calls or online forum posts, as seen in the case of the Reddit user.

Finally, we can speculate about issues of trust stemming from malicious activity. If notifications are viewed as trusted communications from an ISP, bad actors may try and spoof them or trick users into acting on fake notifications. We discuss this issue in depth as part of our threat model in Section 4.1.3.

3.1.5 Quality of threat data

Notifications only work if you have something to notify your customers about. Crucially, this *something* is often derived from various sources of threat data. Van Eeten's evaluation of the Dutch AbuseHub threat sharing platform [23] notes the impact of better data on the Dutch ISP world.

[3] Anecdotal evidence points to another concern: ISP liability if a suggested action goes sour on a customer's system. Section 5.1.1 discusses changing incentives to notify in more detail.

With 90% of Dutch ISPs getting rich data from AbuseHub, their situational awareness greatly increased.

While many ISPs have rich, internal sources of data–and relationships with external stakeholders–the quality of this data isn't always as good as it could be. Uptake of a U.S. government platform has been slow [41], plus access and data expertise may be a challenge for some smaller ISPs [66].

3.1.6 Evaluation and feedback

Evaluations of a notification system's performance is crucial, especially if it's used within a security context. ISPs have not visibly prioritized evaluation and feedback as part of existing notification systems. Some, like Japan's Cyber Clean Center, seem to keep track of basic metrics like re-infection rates[17]. However, other metrics like notification rate, remediation follow-through, ignored notifications, or efficacy by notification channel need to be collected (if they are not already). Several reports, including the ABC for ISPs "Barriers and Metric Considerations", AISI evaluation, and RFC 6108 suggest specific metrics and feedback mechanisms [28, 13, 66].

3.2 Analogous challenges

Identifying these challenges is easy but solving them is a bit more difficult. In this section, we draw some inspiration from similar challenges in the computer security and usable security domains and show how they could contribute to future notification systems.

3.2.1 Warnings and risk

Creating a good notification can be viewed as a challenge in creating and communicating a good warning. The study of warnings is often put under the field of *risk communication*. These communicators have studied how to *extract and target* relevant information to non-experts, as well as how to *frame* that information to emphasize and successfully communicate relevant risks [45].

The risk communication literature is a core part of the *usable security* community that deals with problems closest to ours. Bauer et al. [4] review this literature and create a set of guidelines for creating computer security warnings, some of which inform our guidelines in Chapter 4. Bravo-Lillo et al. [8] run lab studies to probe users' mental models of warnings and also provide relevant design suggestions. Felt et al. [30] and Fagan et al. [26] work on improving warnings in two contexts–SSL warnings and software update messages. Both studies provide valuable examples of how the usable security literature can apply to complex, context-dependent security notifications.

Finally, risk communicators also have a rich history of studying when communications may backfire. Stewart and Martin [63] demonstrate the wide applicability and longevity of warning research. Their 1994 article examines many of the same factors we concern ourselves with, but apply them in the context of consumer product warnings. Notably, they discuss some of the pitfalls of warning design including ineffectiveness, selective attention to messages, and unintended "reactive behavior." This tradition continues–Wash et al. [68] analyze the unintended security consequences of software update warnings and provide a cautionary tale about how poorly designed notifications can backfire.

3.2.2 Notifications

So far, we have implied that the end users targeted by our notifications are *non-expert* end users. This complicates the task of creating notifications as users may not have the requisite security or computing expertise to fully understand a security risk. However, we can still draw lessons from a very similar line of work: security notifications for administrators and expert users.

In two papers, Stock et al. [65, 64] discuss the mechanics of "web vulnerability notification" targeted at maintainers of websites determined to be running vulnerable software. The authors construct a notification system, run trials, conduct interviews with affected parties, and use follow-up scans to calculate response and remediation metrics. They note difficulty with email-based notification, find a relationship between sender trust and remediation, and discuss the feasibility of automated notification. Cetin et al. [12] and Li et al. [39] perform similar pilot studies and both make complementary observations. However, Cetin et al. and Li et al. also go into more detail about the challenges of targeting notifications to the right recipient and providing incentives to remediate. All of these studies provide lessons that are extremely useful for the consumer notification context.

3.2.3 Remediation

Compared to notification, a relatively small amount of work has been done about consumer remediation methods. However, two recent studies provide important lessons. If we view remediation advice as a class of security advice, Fagan and Khan [26] provide an in-depth study of why users may–or may not–follow security advice given to them. By analyzing user perception and motivation, they provide a blueprint for creating effective remediation strategies. Cetin et al. [11] conduct one of the first studies of a specific ISP remediation practice–the walled garden. By studying re-infection rates and user reactions to these walled gardens, the authors find the practice to be "effective and

usable", producing "positive learning effects" in the majority of cases. The authors provide insight into user behavior around a common ISP notification and remediation method. They also demonstrate how a user study can be conducted to gain feedback about a remediation practice–a valuable and necessary piece of knowledge for evaluating a notification system.

3.2.4 Verification

Needing to verify the source, authenticity, or integrity of a message is a common problem. Cryptography is often proposed as a solution. Basic primitives like hashing are effective at verifying integrity, end-to-end encryption maintains integrity, and public-key cryptography help ensure source and authenticity.

However, these are no perfect solutions. A long line of research has demonstrated usability issues with using public-key encryption systems like PGP to encrypt and verify messages [71, 54]. Clark and van Oorschot [14] note the use of SSL over HTTP (HTTPS) for the security and integrity of web requests has largely been successful, but there are still concerns about SSL's certificate authority model, and about the comprehension of SSL errors by end users [30].

We also see new verification methods that stem from new use cases. Encrypted messaging apps have turned to simpler verification methods. Signal pioneered the use of "safety numbers" and scannable verification barcodes, allowing for out-of-band integrity and sender identity checks [60]. These techniques may be the most applicable to the notification context, but should be carefully implemented. Studies by Bicakci et al. and Vaziripour et al. [6, 67] find that Signal's approach is promising, but requires more refinement to help users understand the necessity of verification.

Finally, we can also view the issue of verification through a different lens–phishing. Avoiding malicious notifications is a similar task to avoiding phishing emails and a rich history of work on phishing demonstrates helpful interventions that could be transferred to the notification context. Dhamija et al. demonstrate the effectiveness of visual deception [22] and suggest the use of dynamic interface "skins" [21] as a potential solution. Egelman et al. [24] show how implementing risk communication techniques in the phishing context can help users avoid malicious messages and Sheng et al. [57] show a potential model for anti-phishing education. Finally, Canfield et al. [9] study the behavioral underpinnings of phishing and characterize detection as a "vigilance task". The authors also suggest potential behavioral interventions and security metrics that may be useful.

Chapter 4

Considerations for Notification Design and Delivery

This chapter addresses a simple, yet crucial question: *how should a notification be designed and delivered?* An effective attempt at notification assumes that a *relevant message* is *delivered to the right party*, with *actionable content* that *is not malicious in nature*. These criteria closely relate to the design goals listed at the end of Chapter 1: feasibility, trust, and usability. This chapter begins by looking at security and considers the potential for malicious notifications. Subsequent sections lay out guidelines for the design and delivery of notifications.

4.1 Threat models and trust

Consideration of security is a crucial part of any sort of system design. Our notification system is no exception. Why? Regardless of the underlying technical architecture, a notification system is a new system being overlaid on top of an existing, complex one (an ISP's network). It is also exposed to (and interacts with) the wider Internet, an open system that is not known for its inherent security.

In addition, special sensitivity is warranted because our notification system may be used to communicate security data to end users. Branding of the notification system as a *trusted* conduit from a consumer's ISP requires some minimum level of trust. Because of that trust, it may be tempting for malicious actors to try and co-opt a notification system for their own ends. In this section,

we construct a *threat model* and formally consider certain types of attacks against a notification system.

4.1.1 What is threat modeling?

One commonly accepted (and highly recommended) way to reason about the security properties of a system is to construct a threat model. While there are many approaches to threat modeling, we use Shostack's [59] standard four-question framework:

1. **Characterize the system**: What are we working on?
2. **Identify threats**: What could go wrong?
3. **Choose mitigations**: What are we going to do about it?
4. **Check yourself**: Did you do an acceptable job of the first three steps?

4.1.2 System characterization

We begin by characterizing our notification system. By ensuring we include crucial data flows and user-system interactions, we are able to abstract away underlying implementation details while retaining enough detail to reason about security.

Figure 4-1 lays out an abstraction of a scheme that incorporates *creation and delivery of notifications*, along with *remediation* and the potential for *subsequent feedback*. Direct data flows related to creation and delivery of notifications are marked with solid arrows while secondary data flows are marked with dotted arrows.

This characterization emphasizes the three main stakeholders in the notification process: external groups (red), ISP controlled (blue), and user/consumers (grey). Here, we situate ISPs at the center of the process due to their advantageous position in the ecosystem. Their privileged, upstream location allows them visibility into threats. In addition, their existing trusted relationships with consumers and external stakeholders means they are well-positioned intermediaries to handle the notification process.[1]

Our characterization also opens the door to use of the notification scheme for arbitrary messages not just limited to security. For example, ISPs could use a notification system to programmatically communicate billing, legal, or quality of service information. Right now, these elements aren't usually communicated in a standardized manner (perhaps besides email). In addition, the feedback

[1] In practice, other groups like corporate/enterprise IT departments or managed security providers could also function as an appropriate intermediary.

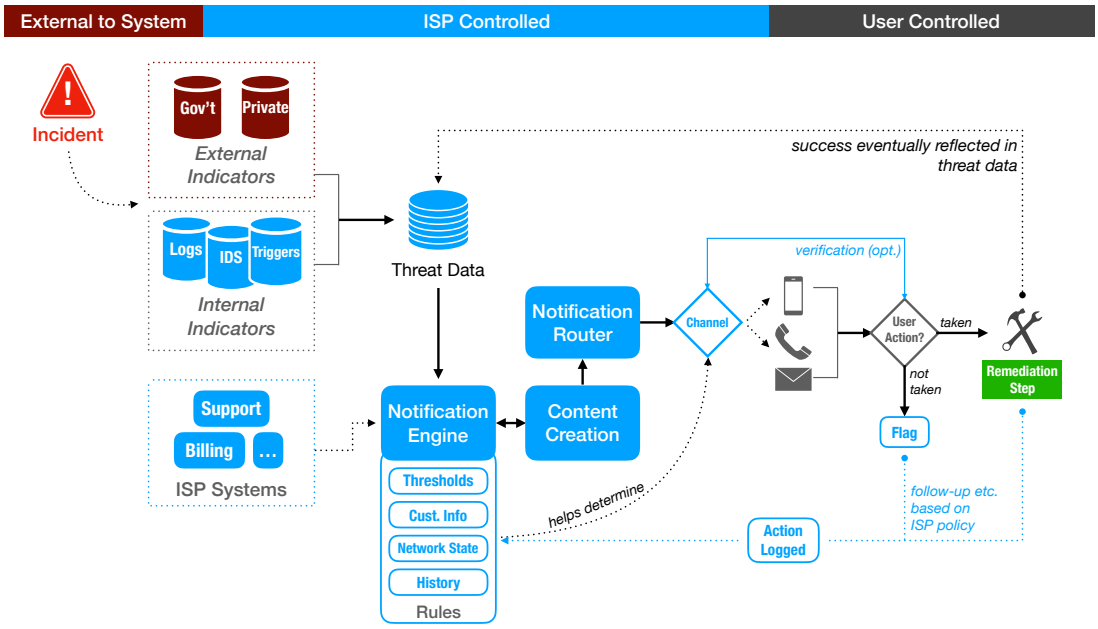

Figure 4-1: Sample notification and remediation flow.

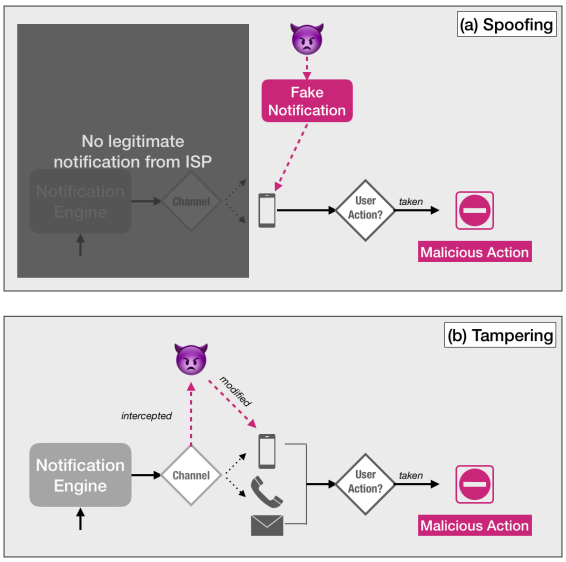

Figure 4-2: Comparison between spoofing and tampering attacks.

provisions in the notification system (e.g., action logging or flagging) could allow for two-way communication. This channel could allow for communication of support (e.g., diagnostics) or regulatory (e.g., quality of service or speed tests) data.

To simplify our threat model, we will fix several assumptions.[2] First, communication between the ISP and external systems (*"external indicators"*) is read only. Second, an ISP has control and visibility into the working of its internal systems (those in blue). Finally, communication between ISP and user controlled systems is done through a defined messaging interface that does not allow for arbitrary code execution (e.g., an idempotent REST API).

4.1.3 Threats

Now that we have a working model of the system, we can turn to our second question. Shostack suggests the examination of six potential "STRIDE" threats: spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege. We focus on spoofing and tampering in detail due to their sensitivity.

[2]We believe these are reasonable: threat data feeds are often subscribe-only, ISPs should have control over their systems, and it is possible to secure a REST API from remote code execution if it is well designed.

Spoofing and tampering

At their core, notifications are about their content –the message that they are trying to communicate. This means that *spoofing* of and *tampering* with a notification's content are especially important threats to pay attention to. Incorrect, misleading, or malicious content could compound or create problems, including existing security issues.

Spoofing of a notification is defined as a malicious attempt to mislead an end user by presenting them with a fake notification that attempts to appear real or legitimate. Attackers may be tempted to do this as notifications will likely be seen as privileged or trusted, especially if they are from a trusted source like an ISP.

Tampering with a notification is a similar, but distinct threat from spoofing. Instead of creating a fake notification from scratch, a malicious actor tampers with the content of a legitimate notification to mislead the recipient. In the case of a security notification, remediation instructions may be removed to prevent removal or a threat or delivery may be blocked to ensure a user stays unaware of a security problem. For non-security notifications, we can envision a variety of attacks. If users expect legal and billing notifications through a system, a scammer could alter a payment link or get a customer to call a fake support number. These are not theoretical attacks; both have been seen in the context of credit card fraud in the U.S. [29].

Mitigation: We detail our proposed *multi-factor verification* solution later in this chapter.

Repudiation

Repudiation threats occur when a system needs to prove that a user performed some action, but is unable to. Repudiation threats may occur if a notification system is also coupled with a logging system for enforcement; for instance, a requirement that users *must* remediate a security issue and prove they have followed the ISP's advice.

Mitigation: Since this isn't the characterized system's primary purpose, this threat is deemed mostly out of scope. (However, logging provisions built into the system could easily be expanded to support nonrepudiation, if needed.)

Information disclosure

Information disclosures occur when data is leaked or released in an unwanted manner. Aside from data in other ISP systems (which is out of scope), the primary data to be leaked is the content of the notification itself. However, we can't claim to know all of the potential privacy harms that

stem from disclosure of notification content. A long-lived attacker could theoretically piece together sensitive information about devices on a network, security weaknesses, and even user activity.

Mitigation: We believe that standard technical measures for data in transit, including encryption, will prevent disclosures. However, the potential for privacy harms stemming from notification data is an important one and should not be ignored. While a privacy impact analysis is out of scope for this thesis, we believe that potential privacy harm should be investigated in future work (see Section 6.2).

Denial of service

Denial of service occurs when an attacker overwhelms a system with requests for service, thereby using up system resources and preventing it from serving legitimate users. A notification system's public endpoints could be subject to denial of service attacks preventing it from creating or sending notifications. On the other side, a broken or hijacked system could flood its users with notifications, causing them to ignore the service entirely.

Mitigation: ISPs should be able to keep most of a notification system off the publicly addressable Internet. In addition, the system should only need to expose one or two endpoints to the broader network to communicate with users. Modern denial of service protection techniques [35] can be used to protect these endpoints and mitigate potential issues. Finally, notification-sending functionality should be designed with DoS prevention in mind including batching of messages and rate limiting.

Elevation of privilege

Privilege elevation happens when someone is able to gain a higher level of authority in a system that the one assigned to them. A classic elevation attack on an operating system might involve a normal user gaining administrator privileges or breaking out of a sandbox. We see no elevation threats inherent to the notification system. Poorly engineered client or server software could expose their hosts to an elevation attack, but that is an implementation-specific threat. Social engineering over notifications (see *Spoofing*) could be seen as a form of elevation, but this is already covered.

Mitigation: Out of scope.

4.2 Form and content: design guidelines

Design is an important element of any type of communication. The need for careful design is especially apparent for notifications as they may communicate important information that, ideally, is understood and acted upon by the receiving user. However, we are faced with an issue: specific communications are usually designed around their content. How can we decide on design guidelines before we're sure of what people are being notified about?

One solution is to view this question as part of existing work on risk communication and the design of warnings. This is a valuable approach as the field of risk communication concerns itself with many similar problems (see Section 3.2). The usable security and human-computer interaction communities have long been in conversation with academic work on risk communication, which further allows us to leverage existing best practices. This section lays out 7 design guidelines inspired by work from Morgan et al. [45] on risk communication and Bauer et al. [4] on warning design.

[R1]: Notifications should be concise and precise.

Bauer et al. note that "[warnings] always interrupt the user", no matter the context. Put another way, users of a system are trying to accomplish the task; a notification, warning, or other interruption of their task flow is usually unwelcome. Brevity and precision will help clearly communicate the purpose of the notification–and thus interruption–to the user. Use of existing risk communication language standards is recommended.[3]

[R2]: Notifications should aim to offer extra context when possible through *progressive disclosure*.

Should I fix the problem with their device now or later? Do I to check their billing statement or can the alert be safely ignored? While notifications should be concise, questions like these usually require context before a decision is made. The progressive disclosure pattern [47] refers to the inclusion of additional features or contextual information which is hidden by default, but easily accessible by users. Well-designed notifications can use this pattern to include links to outside information, provide extra technical detail, or explain why someone is being notified.

[R3]: Security and other risk-based notifications should clearly identify risks and harms.

[3] Many operating systems provide guidelines for alerts and warnings in their developer documentation. Chapter 6 of Morgan et al. [45]'s book also discusses a more comprehensive set of principles for choosing message content.

If a notification is used to communicate a security risk (e.g., a botnet infection) or other risk (e.g., legal action), the risk and potential harms should be immediately apparent to the recipient. The need to make guesses or inferences about risks should be minimized to the greatest extent possible.

[R4]: Notifications should have a consistent design and visual language while respecting interface guidelines of their host platforms.

If ISP-driven notifications become widespread in their use, users will encounter many over the course of their Internet usage. Consistency will help educate and familiarize users, creating training effects which can be leveraged by the broader industry (see Chapter 5, R6). Respect for platforms' native interface guidelines builds on existing familiarity with warnings in technical contexts.

[R5]: Notifications should respect users' time and attention.

As mentioned in R1, warnings are interruptions. A flood of notifications or warnings can backfire, leading users to a state of habituation or annoyance [52, 4, 63]. Because of this, notification systems should consider ways to reduce the burden of receiving notifications. This may include prioritization, labeling, choosing when (not) to proactively alert users about new notifications, and grouping of messages.

[R6]: Notifications should consider the relationship between their message and intended audience.

An additional challenge for ISP-driven notifications is uncertainty as to who will receive them. For example, an ISP may only know about a billing contact and choose to notify them about a security incident. However, the ISP has no way of knowing that someone else in the household usually administers the home's network. ISPs also have no way of knowing about the technical expertise of the user receiving a notification, even though expertise plays a large role in the interpretation of warnings [8]. Because of this, ISPs should consider the desired outcome of their notification and prioritize the creation of clear, targeted notifications.

[R7]: Notifications that suggest remediation should provide meaningful and actionable options.

If a notification suggests a remediation, the options presented should be meaningful and actionable. Users should not have to guess which option is the safest or least destructive and any potential side effects of a remediation should be apparent. Unless safety and integrity are assured, remediation actions should not be executed automatically or taken by default. Regardless of intent, users own

their systems and should not be exposed to unnecessary risk by default.

Sidebar 2 - Good and bad design

Figure 4-3 demonstrates how recommendations may be broken by poorly designed notifications.

- **(a)** ignores **R5** and floods users with notifications. It also gives equal weight to information of differing importance.
- **(b)** is precise, but doesn't highlight any risks or harms **(R3)**, includes too much technical context upfront **(R2, R6)**, and doesn't provide actionable remediation **(R7)**.
- **(c)** is concise and precise, but provides no context or indication of risk **(R2, R3)**.
- **(d)** is actionable, provides a link to extra information, and clearly demonstrates why it's important–but it ignores part of **R7** by automatically taking a destructive action.

In contrast, Figure 4-4 suggests an interface that addresses these issues. The initial notification is precise [(a) and (d), **R1**] but provides extra context through progressive disclosure [(b) and (e), **R2**.] The message's relative importance is indicated and categorized and risks are made apparent [(c) and (d), **R5** and **R3**]. Finally, a meaningful–but non-default–action is provided ((f), **R7**) in a clear manner that adopts the standard user interface guidelines of the host operating system (**R4**).

4.3 Delivery: trust and integrity

As mentioned previously, spoofing and tampering are major concerns. They directly impact the integrity of delivered notifications and successful attacks will likely reduce long-term user trust in the system.

Attacks on delivery and integrity already exist in similar systems. Attackers spoof the visual language of software update prompts [26] to trick users into installing malware. Similarly, phishing emails and websites have imitated legitimate, trusted user interfaces to mislead users and trick them into disclosing sensitive information or performing unwanted actions [22]. Finally, scammers have begun to spoof operating system warnings and error messages as part of a class of tech support scams. These try to convince users into believing that their devices are broken and can only be fixed by support services that will proceed to install malware or steal personal information [32].

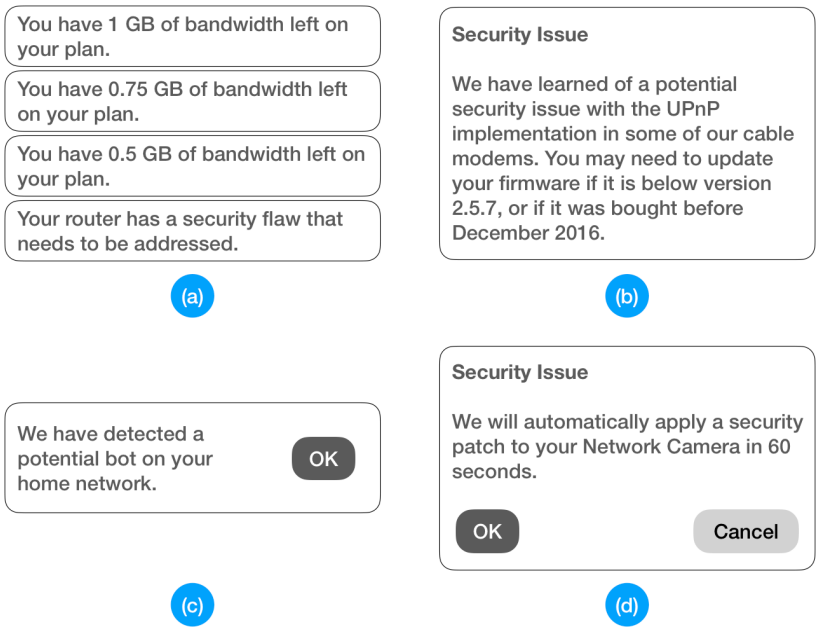

Figure 4-3: Example notifications that break one or more recommendations.

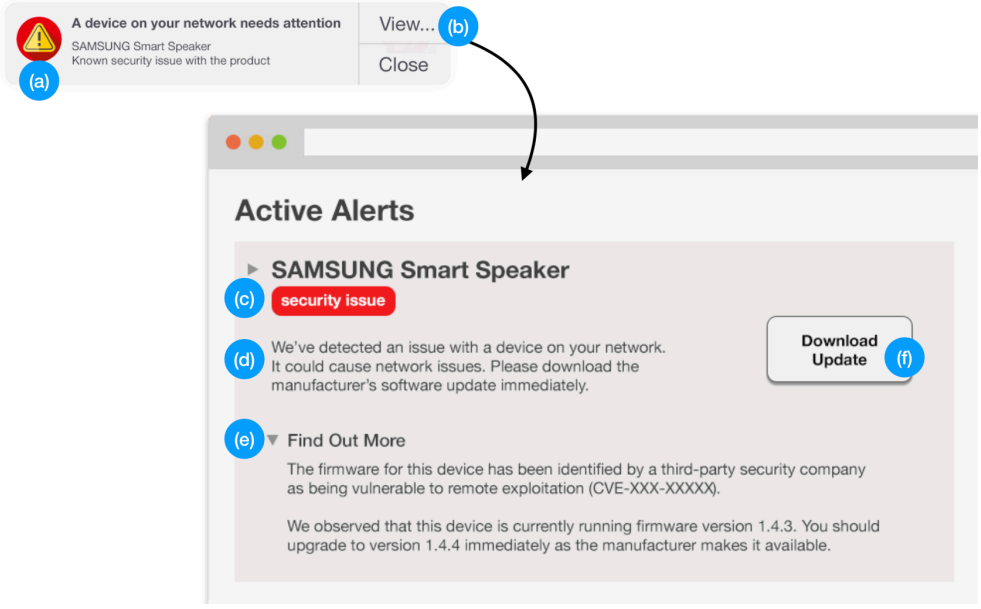

Figure 4-4: A potential design that fixes issues highlighted in the previous figure.

4.3.1 Multi-factor verification

How can we ensure that users maintain trust in notifications? A simple solution would be to solely rely on technical measures such as public-key encryption. All modern web browsers support the HTTPS protocol and alert users of potential integrity attacks. However, it has been shown that users may ignore or misinterpret these warnings [30], even with marked improvement in their usability and design [52]. Poorly designed security warnings may may also habituate users and lead them to ignore valuable information about threats to a system [52].

Instead of relying on warnings, we propose a *multi-factor verification* (MFV) technique. In MFV, users have the option of verifying a notification using at least one method that is *decoupled from how the notification was received*. This technique is inspired by existing best practices in the financial industry. Instead of trusting unsolicited "notifications" from their banks, consumers have been educated to verify the validity of messages by checking other trusted sources (e.g., a bank website) or proactively verifying with the bank (i.e., calling a trusted phone number).

It is important to note that not all notifications will need to be verified. Users may not want to burden themselves with constant, proactive verification. Some notifications, like informational alerts, may not be sensitive enough to warrant consideration. However, we still want to preserve the *option* of verification for when it's needed. Therefore, we suggest that ISPs consider potential spoofing threats for specific types of notifications that they will be sending (e.g. quality of service vs. security) and provide appropriate verification options. Figure 4-5 illustrates this by suggesting how we could create a spectrum of verification options.

[R8]: Notification types that have a high chance of being spoofed or co-opted for malicious reasons should include ways for end users to verify them.

[R9]: ISPs should establish appropriate threat models for commonly-sent types of notifications.

Sidebar 3 - How should you verify?

How should verification be done in practice? What "factors" are appropriate? While more inquiry is needed into the usability of multi-factor verification (see Appendix), this sidebar provides some suggestions.

Passive verification Passive verification allows for users to quickly check integrity without taking much extra action. Since most users own and maintain multiple devices, this property

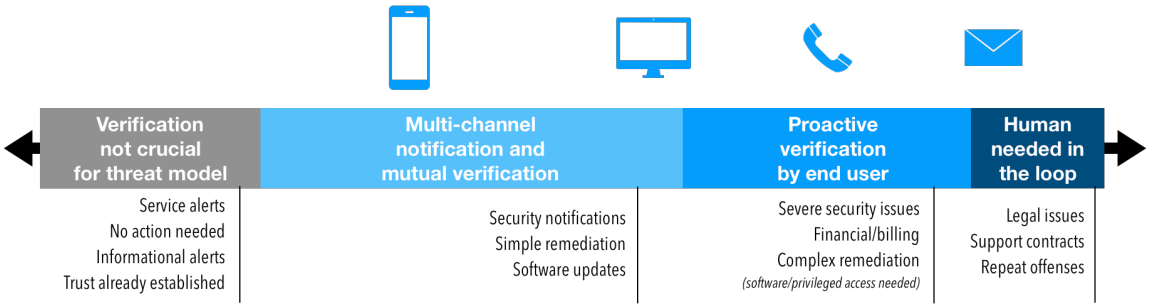

Figure 4-5: Spectrum of multi-factor verification options depending on notification's threat model.

of today's home networks can be leveraged for passive verification. By routing notifications to multiple deices (e.g., a smartphone and a laptop), we can add assurance for relatively little cost. By adding redundancy, users are able to compare received notifications between devices. In addition, this adds an extra measure of protection against spoofing, as it takes much more effort to simultaneously fake a message on multiple platforms.

In addition to multi-device routing, implementations could also leverage existing work on phishing. Dhamija and Tygar [21] propose dynamic interface "skins" that include pre-chosen images or phrases, along with the visual equivalent of a session hash that can be independently computed by both clients and servers. This concept could be extended through the use of newer techniques like trusted execution environments [31] to ensure a UI is not tampered with.

Proactive verification This style of verification requires the user to actively verify a notification. It may be reserved for more sensitive notifications, including dangerous or destructive security remediation (e.g., firmware updates or code execution), or communicate about billing or legal issues.

In the simplest case, this may involve calling a pre-determined, trusted phone number. A user could have the option to call a phone number on their ISP's billing statement to verify the content of a message. Assuming that ISP equipment is trusted, ISPs could also use customer premise equipment (e.g. modems or routers) as an independent notification endpoint and point of comparison. This option could also be combined with a knowledge-based verification technique [36]–including information that only the user and ISP would jointly know. Finally, proactive verification can be supplemented with costlier, established channels of trust like certified mail or human-in-the-loop conversations with ISP staff.

4.3.2 Technical measures

[R10]: Systems used to create notification content should communicate with other systems using end-to-end encryption, where feasible.

A notification system might communicate with many internal and external systems to generate a notification (see Figure 4-1). Each of these connections presents a potential attack vector. While many are outside an ISP's control, software engineering best practices can help us prevent unwanted information disclosure, maintain notification integrity, and boost trust in notification content.

[R11]: Notifications should be logically separate from other internal ISP systems.

More specifically, notification systems may interface with other, sensitive ISP systems like billing or technical logging services. A poorly engineered integration could create new vulnerabilities in existing systems.

[R12]: Notification content should be encrypted in transit between the ISP and consumer, when feasible.[4]

Finally, notification systems should use long-standing technical best practices to maintain the integrity of their content.

[4]Obviously, some notification methods–like phone calls or normal text messages–cannot be encrypted in the traditional sense.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 5

Notifications in Practice

This chapter examines concrete considerations for implementing a notification scheme. It addresses issues beyond the framework-level design considerations in the last chapter.

The discussion in this chapter is grounded in a generalized notification scheme, assumed to be administered by an Internet service provider (ISP) within the United States market, primarily for the purposes of security notification. We begin by briefly reviewing the status quo in the U.S. context. Next, we highlight relevant findings from our framework and background literature. Finally, we detail 10 recommendations for implementation.

5.1 Status quo

In the U.S., the current security notification model centers around ISPs and their customers. ISPs may gather security-related data from their partners and outside sources, both public (e.g. DHS AIS, Spamhaus) and private (e.g. corporate security contracts). That data is usually then used internally by the ISP to determine whether a notification should be made. If there is reason to notify a customer, ISPs will have their own internal policies as to how that notification should be carried out.

Concrete policies from ISPs on when to notify are sparse and hard to find. As of writing, we have not been able to find any public statements on how any ISP in the U.S. market chooses when (or when not) to notify a customer. However, Comcast's Xfinity subsidiary publishes a brief summary of their Bot Notification Policy [72]. This document does not give baseline notification criteria, but

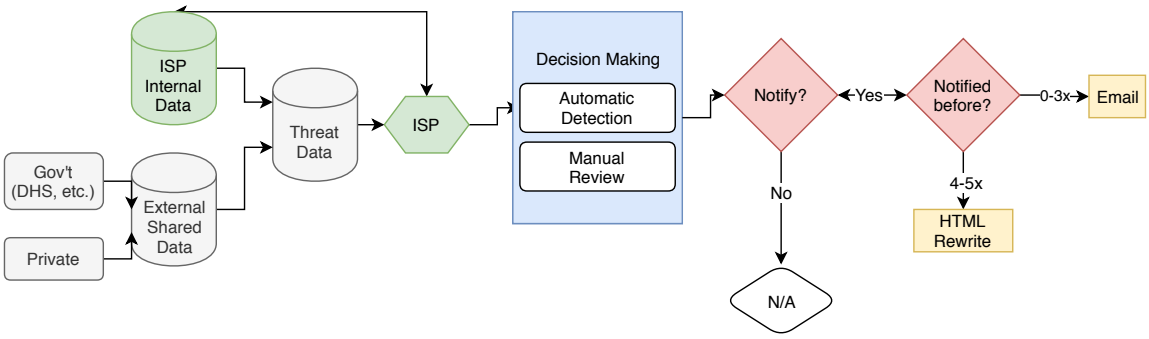

Figure 5-1: Derived flowchart for notification decision making for a large U.S. ISP.

details their policy regarding which notification channel to use. We can also observe the results of interviews conducted as part of an evaluation with the Australian Internet Security Initiative (AISI). Employees of Australian ISPs discuss their notification policies, with roughly half alerting their customers all "all reported daily and/or repeated sighting compromises". The remaining half of ISPs first used manual or automatic "cross-checking", "prioritising", or other internal indicators before creating an alert or sending a notification [66].

While there may be variations between ISPs, we can assume that due to Comcast's position in the market, other ISPs' policies will have at least a passing similarity. We can also assume that, due to its maturity and relative similarity to the U.S., the high-level notification policy decisions seen among Australian ISPs are a good approximation of the decision making processes occurring in U.S. ISPs. Figure 5-1 provides a notional diagram of what an ISP's decision making process may look like. Here, an ISP ingests data from internal and external sources, uses manual and automatic thresholds for triggering a notification, and decides on a channel based on severity and success of previous notification efforts.

5.1.1 Incentives to notify

Currently, ISPs are the main stakeholder with an incentive to notify their customers. If an ISP knows about a security issue on a customer's home network–whether it be spam production, the presence of a botnet, or a visible server misconfiguration–it has several reasons to take action. First, an ISP wishes to keep its own network clean and free of problems; remediating an issue with one of its customers is a relatively easy way to do so. Second, relationships within the Internet operator community are often built on trust and concepts of good citizenship. An ISP's business

relationships (and its employees' personal relationships with colleagues) are strained if its network is acting as a "bad citizen" within the larger community context, potentially impacting the ISP's ability to do business. Finally, ISPs are often the most obvious target of industry self-regulatory or government regulatory action. Legally binding rules or industry codes of conduct can incentivize security cleanup–and hence notification–actions. Asghari's 2010 thesis (see [3]) bears out many of these observations and provides empirical evidence of ISPs' role as an effective intermediary in fighting botnet related security problems.

Changing incentives

Finally, we must think more broadly about how to maintain incentives to notify. One common critique of the ISP as notifying party brings up the theoretical support cost of dealing with notifications. If users get bombarded with notifications and other alerts from their ISP, won't they tie up the ISP's support channels with confusion about the notifications and questions about how to remediate their issues? While informal discussions with employees at a major U.S. ISP indicate otherwise–their company does not view this as a major stumbling block–care must be taken to not make this criticism a self-fulfilling prophecy.

More effective notifications at a higher volume will definitely cause confusion depending on how they are presented. As mentioned in the previous chapter, those designing a notification system should carefully consider content, timing, and audience to not overwhelm customers. Content should provide an obvious path forward and not leave the customer in a confused state. Notifications should be aggregated and prioritized when feasible to prevent "over-notification" and should aim to reach an audience that is able to understand and follow up with any desired remedial action.

Finally, new developments in technology and regulation may spur other stakeholders beyond ISPs to notify. For example, recent proposals for *device manufacturer liability* when security failures occur may push manufacturers to notify their customers of vulnerabilities or security concerns separate of ISPs [18]. Similarly, new vulnerability scanning technology making its way into the consumer market may turn notification into a third-party feature independent of ISPs or manufacturers. For example, some routers and IoT hubs now promise to notify users of security vulnerabilities and needed updates for devices on the network (see Sidebar, Chapter 2).

5.1.2 User experience

Currently, the user experience around notification is wildly varied. Generally, customers may receive notifications via phone, email, HTML rewrite, or captive portal. This uncertain and often disjoint

set of notification channels is not ideal. This uncertainty is compounded by the issues of trust and notification perception raised in Section 4.1.

It is important to note that these notifications are not ineffective; undoubtedly, many customers and users appropriately respond to notifications received by email, phone, HTML rewrite, and captive portal. As mentioned previously, Comcast's HTML rewrite architecture has been codified in RFC 6108 [13] and may have been adopted by other ISPs and organizations. However, due to a lack of evaluation data, we have no way of knowing these methods' true efficacy. Needless to say, though, there is work to be done on the user experience front.

5.1.3 Sourcing threat data

Where do ISPs usually get data on threats? Most rely on a variety of sources. These include sources internal to the company, data from private security companies and industry partners, and government-run data sharing platforms. Information on threat data from internal sources and industry partners is hard to come by, usually for security and contractual reasons. However, we are able to discuss government-run data sharing platforms in more detail.

In the U.S., the Department of Homeland Security (DHS) administers the Automated Indicator Sharing (AIS) system for the exchange of threat data. However, evidence has shown that AIS has not been widely adopted by industry. Many stakeholders in the cybersecurity industry have noted the AIS data's lack of context and DHS' tendency to over-classify certain types of data [10].

An alternative to AIS' broad threat sharing approach is one that emerges from the Australian Internet Security Initiative (AISI). The AISI acts as a trusted intermediary, fusing both private and government data sets for use by its ISP "customers". Relevant data is then fed to ISPs who can prove ownership over certain IP address blocks. This tailored approach addresses many of the issues with AIS, including the lack of context. It also side-steps the classification issue, with aggregation and transformation of classified data happening before the threat data is shared. Streamlining this data sharing process for ISPs can provide great benefit, as it enriches ISPs' knowledge of the threat landscape and allows them to address a greater number of potential threats to their customers.

5.2 Recommendations: improving the model

In previous sections, we have established the status quo for notification schemes in the U.S. and identified several key areas for improvement. Now, how might those improvements be implemented?

This section lays out recommendations for implementation based on the framework in Chapter 4 and the improvements identified in 5.2. It does so by addressing three broad areas: suggestions for institutions, creation of standards, and handling of consumers. The recommendations keep in mind the fact that the U.S. ISP market is large and dynamic in nature. To that end, recommendations for each area are categorized for short-term or long-term implementation, and do not aim to be overly prescriptive in order to maintain flexibility.

5.2.1 Institutions

The creation and maintenance of a successful notification scheme requires the interaction of many institutions. Chief among them are the ISP and its sources of threat data. This section provides recommendations for both groups.

[R1] ISPs should build feedback and evaluation loops into the operation of their notification scheme.

Long-term. Feedback and evaluation data is key to the success of any intervention. At an institutional level, one of the most pressing problems for the implementation of notification schemes is the lack of feedback and evaluation data about their efficacy. Industry-wide summary statistics may be available about notification and re-infection rates, but anecdotally, many U.S. ISPs do not have easy access to many basic metrics, such as notification rate, remediation follow-through, ignored notifications, or efficacy by notification channel. Knowledge of these metrics will allow ISPs to continually reassess how effective their scheme is, to tweak notification strategies as needed, and to provide better service to their customers.

In fact, the U.S.'s ABC for ISPs cite the need for better metrics in evaluation in a 2012 annex to the main Code [28]. In a similar vein, an evaluation of the Australian Internet Security Initiative recommended research into the perceptions of notifications within the Australian ecosystem [1]. A study was considered but never done [42], indicating the need for such work.

In addition to these basic metrics, we recommend that ISPs consider feedback and evaluation strategies that go beyond the basic metrics described in the 2013 CSRIC WG7 report [28]. As an ISP's notification scheme evolves and matures, the ISP can integrate evaluation data from other operational sources. For instance, an ISP could integrate data from its support agents or other customer-facing feedback mechanisms to understand users' perceptions of the scheme. It could also compare the efficacy of its scheme across market segments, customer bases, or even geographic location. By combining this knowledge with a suitable A/B testing framework, the ISP could even

optimize notification strategies for different customer populations in much the same way that an advertiser adjusts its targeting metrics.

[R2] ISPs should have one person or department "own" responsibility for the notification scheme.

Short-term. Currently, responsibility for notification schemes is often unclear or ill-defined. This makes it difficult for ISPs to engage in discussion about the scheme, difficult for schemes to be administered, and difficult for schemes to be promoted. Centralizing responsibility allows a group or individual to take ownership of the notification process and streamline interactions with other institutions, both inside and outside the ISP. ISPs can also consider substituting centralized ownership with establishing a central point of contact as an even shorter-term measure.

[R3] Data sharing organizations should consider a "lighter is better" mentality.

Long-term. Existing data and threat-sharing capabilities within the U.S. government fall under the Department of Homeland Security and its Automated Indicator Sharing initiative. As discussed in section 5.2, its "omnibus" approach to data sharing infrastructure is ambitious, but may not be the best fit for powering a notification scheme. Instead, we suggest the creation of an overlay system on top of AIS that mirrors the Australian Internet Security Initiative's threat sharing program. This overlay could ingest data from a variety of sources (including AIS), but would have lower barriers to entry for both subscribers and reporters, and would be tailored exclusively to the notification needs of the network operator community.

While this overlay could be operated by DHS, we suggest that responsibility lie with an existing industry group. Recent recommendations from the U.S. Departments of Commerce and Homeland Security [20] suggest a similar structure for information sharing. The report explicitly notes the need for greater inter-industry cooperation between "ISPs and their peering partners" and suggests the federal government "facilitate" this activity through existing domestic and international agreements and data sharing structures. [1]

[R4] A multi-stakeholder process should be convened to continue existing work on notification schemes.

Short-term. The U.S. telecom industry was crucial in the first wave of work on notification schemes.

[1] This suggestion falls under Action 2.1 of the report: "Internet service providers and their peering partners should expand current information sharing to achieve more timely and effective sharing of actionable threat information both domestically and globally." The report's suggested role of government as facilitator points to existing agreements and forums discussed in Chapter 2, such as information sharing and analysis centers (ISACs), as well as other "international peers" (e.g., ISAC Japan) outside the scope of this thesis.

Early work such as the Anti-Botnet Code of Conduct [27] and informational RFCs on in-browser notification from operators like Comcast helped launch the field. However, the public face of this work has stagnated. We believe that the industry should continue to build on this solid foundation of work and convene a multi-stakeholder process in the mold of the Anti-Bot Code's working groups, or more recent convenings by the U.S. Department of Commerce and NTIA. A multi-stakeholder process would allow for the integration of diverse perspectives from industry, government, academia, and consumers. It would also allow those in industry to reinforce existing relationships built during the first wave of notification work. Finally, it would create short-term momentum to continue work on an industry-wide scale.

5.2.2 Standards

As discussed in Chapter 2, there are a diverse set of proposals that deal with notification. However, this prior work is not sufficient for a holistic notification scheme. Recommendations in this section address the advancement of standards work, along with two guiding principles.

[R5] Industry should prioritize the creation of BCPs and self-regulatory standards.

Long-term. As this recommendation goes hand-in-hand with R4's multi-stakeholder process, it can be viewed as its long term equivalent. As the community of practice surrounding notification grows, we believe that industry should target the extension of existing codes and standards (e.g., an update of the Anti-Botnet Code), as well as the creation of new best practices as needed. These best practices can include the design standards mentioned in R6, but could also encompass a wide range of topics including best practices for feedback, the creation of metrics for evaluation, standards for multi-factor verification as discussed in Chapter 4, and the codification of operational best practices for use by ISPs and network operators.

In addition, industry should consider how to turn notification into a common practice for the collective benefit of both the industry and the Internet ecosystem. While there are many ways to achieve this goal, we might look to the success of Australia for inspiration. The Australian Communications and Media Authority (ACMA) has examined the possibility of an industry self-regulatory body to deal with spam [66]. Similar to spam, notification and remediation are a common "public health" issue that all players in the market must deal with. A move towards self-regulation would encourage the development of standards and help a community of practice develop.

[R6] Stakeholders should consider a coordinating design standard for the appearance and verification of notifications.

Long-term. As part of the development of standards, all stakeholders should strongly consider creating a coordinated design standard for notifications' appearance. (See Section 4.2 for more on the necessity of design standards.) Maintaining a uniform face for notifications across different ISPs and different contexts will allow the industry to leverage training effects across their customer bases. It may also reduce uncertainty for users new to alerts. Put another way, consumers are familiar with common labeling in other contexts. How confusing would it be if nutrition facts were laid out in a different format by company?

By specifying a common design language in the standards process, stakeholders can help reduce future confusion when a customer switches ISPs and increase familiarity for all consumers across the industry. Extending that same language for the whole lifecycle of interaction, including verification, helps boost the user experience and overall efficacy of any notification scheme. Design standardization also allows for discussions about accessibility and internationalization to occur. Making notifications available in different languages and ensuring that they can be universally received and understood is a good design practice, as well as a good business practice.

[R7] ISPs should ensure the architecture of their notification scheme can be generalized to different threats and contexts.

Long-term. While this recommendation mirrors an existing discussion in the Anti-Botnet Code, we believe it bears repeating. Many of the existing issues with notification schemes stem from the fact that they were standardized through corporate policy without an eye towards flexibility. As computing paradigms shift, ISPs need to ensure their notification strategies keep pace. To that end, ISPs should not make assumptions about what type of device or interface will be receiving the notification. For instance, earlier standards make assumptions about the availability of unencrypted Web traffic to inject HTML and JavaScript based overlays; this type of assumption should be avoided. Remaining neutral here will also make it easier to implement suggestions about multi-factor verification, discussed previously in this chapter.

ISPs should also make sure that their notification scheme is not overly tailored to one specific threat or class of threat. Many previous initiatives (see Chapter 2) were scoped to deal with one specific type of threat (e.g., a PC-based botnet) or tailored to a specific incident (e.g., the Conficker worm). This type of scoping can be highly effective, but it does not support the operation of a broad notification scheme. By keeping in mind the various classes of threats that a notification scheme will have to deal with, making generalizability and explicit design goal during the standards process will help ISPs ensure that their scheme stays relevant.

5.2.3 End Users and Consumers

Finally, we should not forget that the success of a notification scheme rises or falls on its interactions with end users and consumers. This section provides three suggestions on how to handle the role of the end user in the scheme's development.

[R8] End users' agency and role in the remediation process should be respected.

While customers of an ISP are subject to contracts and obligations imposed by their customer relationship, operators of a notification scheme should respect their customers' and end users' agency and role in the process. While ISPs have an excellent high-level view of a threat, end users may have a better understanding of local context. For instance, requiring immediate action may be desirable from a systems perspective, but an end user may not wish to interrupt current activity or may be unable to take immediate action. Any user who has experienced the threat of an immediate, uninterruptible software update can recognize this fact. In fact, research has shown that poorly constructed update notifications can cause significant annoyance and may even worsen security in the long run [26].

A core tenet of user-centered design is that the end user is a partner of the system's designer. The user should be brought on board with the design goals of the system instead of being forced into a role. The design and implementation of any notification scheme should respect this view.

[R9] End users' time and attention should be respected.

Much has been written in other contexts about the issue of notification overload (see R4, Section 4.2 or [4, 8]). The operation of any notification scheme should respect the fact that end users' priorities may not always align with that of the ISP or industry; in other words, remediation may not be the user's foremost priority. While policies for notification should encourage remediation, they should also ensure that a balance is struck between urgency and respect for end users' time and attention.

[R10] End users and consumers should be engaged in an ongoing conversation about security and remediation.

Finally, notification may be a new paradigm for many users. Light training and education may be needed to establish new norms of user behavior and interaction with the ISP, especially if the ISP decides to implement multi-factor verification. This also speaks to the need to establish an ongoing conversation with end users and consumers about security and remediation. As the threat landscape evolves and consumer behavior changes, ISPs will need to respond to these changes and

users will need to keep pace.

What form this engagement takes is up for debate. Certainly, we believe that formal user studies are necessary to fully understand how end users interact with new notification paradigms (see Appendix 1 for a potential starting point). These studies are necessary to fully probe users' expectations and understanding of the notification process. They also serve as testing grounds to fully evaluate the security and privacy harms that could stem from building a notification (Chapter 4). However, executing this type of study may not be of interest to (or within the expertise of) an ISP. Other forms of feedback–reviews of evaluation data, focus groups, surveys, and "power user" feedback–are all valid and extremely helpful in this case.

5.2.4 International Considerations

While these recommendations are targeted at the context of the U.S. market, they are broadly applicable to the creation of a notification scheme in many other markets. For example, suggestions for how ISPs should internally handle notifications (R1, R2) and for how end users should be considered (R8, R9, R10) are largely context independent. Similarly, standards work at the design and architecture level (R6, R7) are mostly independent or market or location.

The largest difference between the U.S. and other international contexts is in data sharing (R3) and stakeholder (R4) landscape. While the Internet governance community has used multi-stakeholder processes in a variety of international contexts (e.g., ICANN and the IETF), there will be varying amounts of work and local expertise on the issue of notification schemes. For example, initiatives in Japan, South Korea, Australia, and the European Union meet or surpass existing work within the U.S., so local multi-stakeholder efforts would be able to draw on existing communities like in the U.S. However, more work may need to be done to identify and convene the stakeholder community in other locations if R4 is to be followed. Similarly, both the downsides and benefits of the DHS' AIS are unique to the U.S. context; work will need to be done to tailor R3 to the national or regional cybersecurity governance structures elsewhere.

5.3 Conclusion

Right now the security notification model most U.S. ISPs use is functional, but not sustainable, from both a technical and a usability standpoint. ISPs may have incentives to notify their customers of security issues, but their framing and delivery leave much to be desired. In addition, gaps exist in how threat data is shared with these ISPs from government sources. We propose 10 recommen-

dations for institutions, the creation of standards, and consumers to address these implementation issues. While not exhaustive, these recommendations highlight short-term and long-term avenues for improvement in a generalized notification system for an ISP.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 6

Conclusion

As Internet service providers become more and more central to our online lives, they are well positioned to be an extra set of watchful eyes for our devices and networks. Through their privileged vantage point as an intermediary, they may be able to detect security problems before they spiral out of control–and notify us about what is going on and how to fix it.

However, notifications are deceptively difficult. While they may take the form of an email, phone call, or pop up at their simplest, administering an ISP-scale notification system is a hard thing to do. Numerous questions present themselves to the individual or group creating the notification system, including some of the following:

Where does the data come from? What should I notify people about–and what should I ignore? Where do the notifications get sent? What communication channels should I use to notify my customers? What should the notifications look like? How do I create the textual or visual content inside the notification? How do I ensure the notification gets to the right place at the right time? How can I make users take the correct remedial action if I notify them about a problem? How do I ensure my customers are confident in the validity of the notification? How do I evaluate my notifications to make sure they're working?

All of these questions are tough ones to answer. They involve interfacing with a complex web of internal and external stakeholders that provide data. They touch on deep, unsolved questions from the field of human-computer interaction on users' risk tolerance and risk perception in computer security. They deal with issues of uncertainty in data. Perhaps most importantly, they touch on

the complex Internet usage patterns of an ISP's customers and end users, each unique and full of context.

6.1 Contributions

This thesis provides an answer to many of these questions. By drawing on existing work from a diverse set of disciplines–ISP security, threat data sharing, risk communication, and human-computer interaction–it provides a model for creating a smarter, more usable notification system.

We begin by highlighting lessons from prior efforts in the area (Chapter 2) to learn from the rich history of anti-spam, anti-botnet, and security remediation work that already exists in the academic and operational communities. This prior work is used to identify challenges and pain points in the current notification process. It is also used to highlight potential solutions from other contexts, ranging from the usability of SSL certificates to verification procedures used by the financial industry (Chapter 3). We then provide a threat model and design framework for this system (Chapter 4), allowing potential implementations to be created according to best practices identified by this thesis. This framework contains novel solutions to hard usability problems such as verification of a notification's validity. Finally, we leverage the framework to provide actionable implementation suggestions for ISPs with existing notification systems through a generalized case study (Chapter 5).

6.2 Future work

In addition to questions about the core design of the notification system, the development of this thesis raised several areas for future research. These can be broken down into three broad categories: technical research towards implementing a secure notification channel, testing the operational suggestions in Chapter 5, and more detailed studies of notification usability.

6.2.1 Technical implementation

There is obviously a large gap between a *framework* for the design of a notification system and an *actual*, running notification system. We envision two main pieces of technical work to be done in this area: creation of a secure, ISP-to-customer notification channel and development of a notification endpoints for different devices.

The presence of a secure, ISP-to-consumer notification channel forms the core of any higher level

notification framework. Bauer's Net.info proposal–one of the chief inspirations for this thesis–provides one model of such a channel [5]. The Net.info system envisions a three-part system, including a global resolver, an ISP-hosted notification server, and an endpoint client hosted on customer premise equipment (or a user's device). Clients would look up their ISP's Net.info server using the global resolver, the resolver would point to the ISP-hosted server using a DNSSEC secured pointer, and clients would then subscribe to notifications from that server (see Figure 6-1).

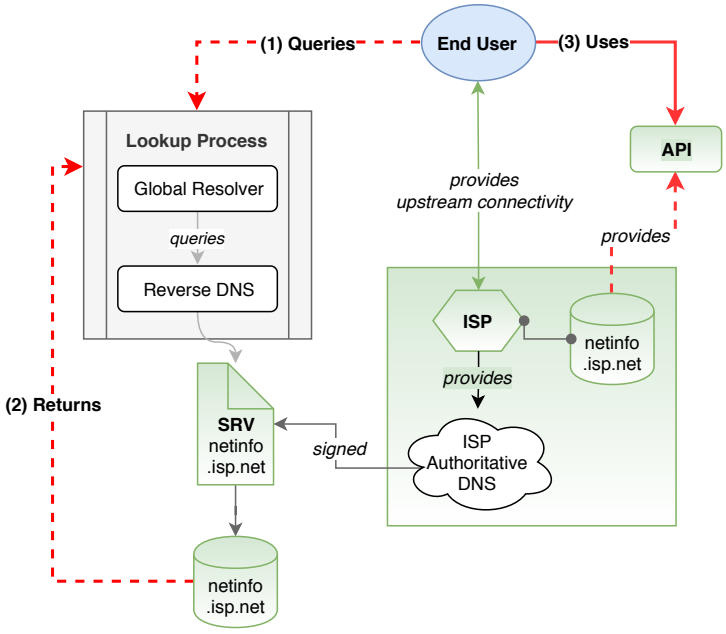

Figure 6-1: Net.info logical diagram.

Work has begun to prototype the Net.info global resolver and ISP-hosted API server, but these prototypes are not yet operational. Furthermore, these prototypes have not been integrated with their user-facing components. We see the following implementation tasks as a series of potential milestones:

1. Create a working prototype of the Net.info global redirect server.
2. Write a proof-of-concept extension to an existing DNS server to create signed, ISP-specific pointers to notification servers.
3. Create a working prototype of the Net.info API server.
4. Prototype notification endpoints, including desktop and mobile clients,
5. Prototype endpoint "gateways" for non-traditional channels, including voice and text

6. Formalize a RFC-style specification for each component of the notification architecture

The creation of any complex system also raises further questions about the system's undefined behavior, security risk, and privacy threat. Once an implementation and specification are created, further research into these issues is also warranted. Relevant questions may include:

1. What problems could an insider threat create for a notification system?
2. What possible avenues of attack are there against the redirect server, ISP server, and client?
3. Can the content of a notification pose any privacy or security risk if the wrong party is notified?
4. Does administration of the redirect server or ISP server create any obligations under privacy regulations like the GDPR?

Net.info is just one of many potential notification architectures. Other systems may be warranted for use of notifications outside of the ISP-consumer context. While not in the scope of this thesis, these alternative systems could find use in other areas, such as software updating or notifications to system administrators instead of end users (see [64]).

6.2.2 Testing operational suggestions

In addition to creating the underlying technical architecture for notifications, there are many practical considerations for implementation within an existing ISP or organization. While we provide a set of targeted recommendations in Section 5.2, they don't capture the full complexity of turning a notification system into an ecosystem-wide endeavor. We foresee many challenges on this front, including:

1. Designing feedback and evaluation metrics for notification schemes
2. Finding an appropriate institutional home for notifications within an ISP
3. Assessing the feasibility and (potential membership) of a multi-stakeholder process on notification schemes
4. Attempting the integration of a notification system into existing ISP processes, like their customer support systems and business service contracts
5. Assessing the applicability of a notification scheme for non-U.S. ISPs (e.g., in the case of a multinational telecom company)

Challenges associated with obtaining data to drive notifications also pose some potential tasks:

1. Testing the integration of existing threat data feeds into the notification system

2. Prototyping an ISP-specific "overlay" system to ingest and package threat data for notification purposes
3. Research into integrating "local" sources of threat data (e.g., from consumer IoT gateways or local routers) into the notification framework

6.2.3 Usability

Finally, we also believe there is work to be done examining the usability of the notification framework presented in this thesis. This work falls into three broad categories: the security notification context, validation of the usability of multi-factor verification, and research into appropriate framing and presentation of notifications.

Security notifications

The trio of user studies presented in Appendix 1 is a first step towards validating the use of novel notifications in the security context. If we assume that the framework presented in this thesis forms the basis of a new notification system, there is much more work to be done within the security context. Ideally, usability should be considered and tested across a wide set of use cases, potential adversarial threats, notification types, audiences, and social contexts.

We envision several potential research questions:

1. How does notification channel affect follow-through in remediation?
2. How might insider threats [not considered as part of the threat model in Chapter 4] exploit the system?
3. How should notifications appear outside the home context? Are they appropriate for corporate or enterprise settings?
4. How do end users react if they receive a notification not tailored to their level of knowledge?
5. How does prior experience with remediating security issues change perception of notifications?
6. How might perceptions of notifications change depending on their stated source?

Trust and verification

The concept of multi-factor verification (MFV) is not new but is novel in the context of verifying the integrity of notifications. Notably, though, the usability of MFV has not been studied in too much detail in its other contexts (financial activity verification and public key verification; see Section 3.2.3). This indicates that studies of the efficacy and usability of MFV are a potential avenue for inquiry outside of the pure notification context.

Research questions may include:

1. What potential channels could be used for MFV beyond the ones assumed in this thesis?
2. What other contexts or business processes can MFV be used for outside of the notification context? Outside of the computing context?
3. Can the MFV paradigm be used to combat phishing?
4. How can end users be incentivized to verify communications, even when they believe a communication to be real or legitimate?
5. Can MFV be made usable for individuals without access to additional verification factors (e.g., no phone)?

In addition to the MFV paradigm itself, it is also worth doing further work on the use of MFV within different business and operational contexts. Research topics may include setting appropriate boundaries for different activities along the verification spectrum and the practical integration of MFV into existing IT and support workflows.

Broader notification context

Finally, the framework presented in this thesis is motivated by security concerns, but should not be viewed as exclusively for security communications. Because the framework can be generalized, there are many questions that should be asked about how notifications are designed.

Potential research topics include:

1. What is the most user-friendly way for notifications to be routed among multiple devices?
2. Are current interaction paradigms for notifications (push alerts, pop-ups, notification trays) sufficient, or should they be changed?
3. What would a standardized, cross-ISP design language look like for notifications?
4. How can notifications support progressive (layered) disclosure of information?
5. What is the best way to create a system where multiple end users receive different notification content for the same event?
6. If a notification requires multiple follow-up steps, what might a "support ticket"-like system look like to track interaction history with a notification event?
7. How can voice UIs (e.g. home assistants) be leveraged as a notification channel?
8. How can the content of one notification be flexibly adapted for multiple clients (e.g. desktop, phone, text, voice) without explicitly designing content for each?

Appendix A

Appendix: Design Validation Study

This appendix lays out a potential user study to validate key design decisions made during the creation of the notification framework.

A.1 Motivation

The framework laid out in Chapter 4 makes several substantial claims about the correct way to design the form and content of a notification, along with appropriately framing the delivery of its message. These claims include ones about trust and integrity, such as the implementation of a two-factor verification system. They also include design related guidelines, such as a "progressive disclosure" model for notification content

While all of these claims and guidelines are based on existing literature from the usable security and human-computer interaction communities, their use within the novel notification context may bears closer examination. Many of these theories–such as mental models-informed design and progressive disclosure–were first studied in other contexts. Other ideas, such as two-factor verification, are novel within the computer security notification context. This leads to questions about these guidelines' efficacy. For instance, does progressive disclosure work for recipients with different levels of expertise? How will notification recipients react to the same message interrupting different types of tasks? More fundamentally, will users actually follow multi-factor verification best practices?

This section lays out a potential user study to probe these questions. While running the study is outside the scope of this thesis, performing such a study on high-fidelity prototypes of a notification system would be a crucial next step in implementing this document's design suggestions.

A.1.1 Related factors

Here, we construct a standard set of survey instruments to be performed by each participant of a user study. This survey will be used to gather information on potential correlates and covariates of task performance, including technical knowledge, security behavior, and risk taking.

Technical knowledge

Baseline technical knowledge, especially knowledge about computing, may influence end user behavior in the notification context. Users with more knowledge will most likely have more complex mental models of security issues, drastically changing their behavior in the face of threats. We borrow the "Internet Know-how Self Report Scale" and "Technical Network Knowledge Scale" from Kang et al. [34] as one set of proxies for knowledge level. Both scales consist of questions that should be answerable by those with the equivalent of college-level knowledge in the area. Both scales have been validated by Kang et al. ($\alpha = 0.79$ and $\alpha = 0.88$, respectively) and was shown to be a significant discriminant between populations in the authors' paper.

Security behavior

To more directly probe participants' levels of security knowledge, we also administer the Security Behavior Intentions (SeBIS) scale as detailed in Egelman et al. [25]. SeBIS contains four subscales, each validated to correlate with a set of security behaviors: *awareness* (security vigilance), *passwords* (good password "hygiene"), *updating* (applying software updates), and *securement* (taking proactive, protective action). Since the scale has predictive ability and behavior-centric subscales, it can help tease out determinants of participant behavior within the user study context.

Risk taking

Finally, individuals' level of risk taking has been shown to be a significant factor that varies along with security behavior. Intuitively, those with higher propensity for risk taking may have lower bars for violating a given threat model. This may have a significant effect on how participants interact with a notification system in wide use; individuals with a propensity towards risk must be accounted for in the design of any system for use by the general public.

To account for risk taking behavior independent of the security context, we include a reduced

version of the Domain-Specific Risk-Taking (DOSPERT) scale [7]. DOSPERT has been validated by multiple authors across different cultures and contexts.

A.2 Hypotheses

We propose three categories of hypotheses related to clarity, trust, and notification audience.

A.2.1 H1. Reported clarity.

Are users able to understand the notifications they receive? Do they understand the appropriate remedial action being requested? Does progressive disclosure help? Clarity and understanding of a notification message is crucial. If a user doesn't understand what a notification is about or what it's asking them to do, the message is not helpful (and may end up being harmful). These lessons have been proven time and time again in various contexts. In the cybersecurity world, we have seen studies of everything from software updating [26] to SSL certificates [30]; these results echo earlier research on product warning labels from almost 30 years ago [63].

We hypothesize that:

- H1-A: Participants with *higher baseline security knowledge scores* will self-report a higher level of notification understanding, regardless of condition.
- H1-B: Participants *in progressive disclosure conditions will self-report a higher level of notification understanding*, relative to those outside of progressive disclosure conditions.
- H1-C: Participants in conditions *with notification content incongruent with their current activity* (see Experimental Design) will self-report lower levels of notification understanding.

A.2.2 H2. Multi-factor verification and trust.

Does multi-factor verification (MFV) work? Are users likely to use MFV? While MFV has been proven to work in other contexts like financial transaction verification, it's uncertain how users will react to this novel form of interaction in the security context. Existing work that studies the adoption of two-factor authentication (2FA) services has shown that contemporary 2FA systems are often more accommodating to users' workflows, but still suffer from negative perceptions of usability [69].

Will this same trend hold for MFV? We hypothesize that:

- H2-A: Participants with *higher awareness and securement SeBIS scores* will be more likely to utilize MFV.
- H2-B: Participants with *lower risk-taking scores* will be more likely to utilize MFV.
- H2-C: Participants with *higher baseline security knowledge* will be more likely to utilize MFV.
- H2-D: Participants who utilize MFV *are more likely to correctly identify malicious notifications.*
- H2-E: Independent of MFV use, those with *higher baseline security knowledge* are less likely to trust the authenticity of notifications.

A.2.3 H3. Audience.

Who should a notification be directed towards? Will they take the correct action? Often, there may be a mismatch between the recipient of a notification and the user that is most equipped to deal with the notification. For example, an ISP may email a security alert to Bob, their billing contact on file. However, the ISP lacks knowledge about other users of the connection beyond the subscriber; it has no way of knowing that Alice is the most savvy user in Bob's household and that she is the family's *de facto* tech support.

These relationships are often difficult to determine for those outside a household or organization. For example, research shows that individuals use a diverse set of criteria for determining hose security advice to follow, making decisions difficult to predict [51]. However, we believe that baseline level of technical expertise provides a good proxy for (1) those most likely to provide support in the home context and (2) those most likely to act on a security notification.

Therefore, we hypothesize that:

- H3-A: *there will be a positive correlation between baseline security knowledge and correct remedial action* across all conditions.
- H3-B: *there will be a positive correlation between SeBIS score and correct remedial action* across all conditions.
- H3-C: Those with *low baseline security knowledge but low risk-taking scores* will likely take correct remedial action.

.

Factor	Level 1	Level 2
Notification Type	Non-Security	Security
Content Length	Full	Progressive
Malicious	Real	Malicious
MFV	Can Verify	Can't Verify
Task Congruence	Incongruent	Congruent

Table A-1: Potential factors for a user study.

A.3 Experimental design

We propose an *experience sampling* methodology [53] to best probe users' perceptions of notifications in the context of normal computer usage activity. In an experience sampling methodology, a participant is periodically interrupted while they are performing an experimental task. The interruption is usually used to administer a short questionnaire, take a measurement, or have the participant perform some other study-related activity.

Event-triggered experience sampling refers to the use of an event instead of a timed trigger; this methodology most closely aligns with our proposed study method. As notifications are usually triggered by external events (e.g., a security event or a push from the ISP), we can substitute the standard questionnaire or measurement for the actual notification in our study. This approach has precedent and has been previously used by Reeder et al. [52] users' perceptions and reactions to Web browser warnings in various natural contexts.

A.3.1 Study Components

Along these lines, our experimental design is broken down into four main components: a cover task, the notification event, response to the notification, and the post-task questionnaire.

Participants will be assigned a **cover task** to complete. These tasks should be simple, open-ended, and drawn from a pre-defined set of standard Internet use activities. For example, a participant could be asked to *send an email*, *purchase an item*, *write a blog post*, or *upload a video*. It is important that this task is unrelated to the study's focus on notifications. Security notifications are often unexpected and remove users from the context of their current activity, and this dynamic should be replicated by the study. The task also serves as an experimental deception to misdirect participants from the true purpose of the study.

At some point during the participant's execution of the cover task, the researcher should trigger a **notification event**. We propose five potential factors to vary across conditions (see Table A-1), creating an upper bound of a 5-factor, 2-level study. This type of factorial study could be run as either between-subjects or within-subjects depending on the hypothesis being investigated. For instance, researchers focusing on differences in notification type may want to run a between-subjects study, but those looking for training effects after repeated notification exposure will want a within-subjects design.

The proposed factors each focus on one key part of the notification design. Utilizing all five factors at once is feasible and leads to a study that can address all hypotheses at once. However, this requires a large number of participants for statistical significance. Given the common rule of thumb of 20 participants per cell, such a study would require $2^5 * 20 = 640$ total participants. It may be more feasible to select a subset of factors and run disjoint user studies to more effectively probe individual hypotheses. For example, a researcher curious about multi-factor verification may want to run a simpler 2x2 study and only vary *Malicious* and *MFV*, holding the other factors constant.

After the notification event, the user will be observed during their **response to the notification**. Participants should be allowed to interact with the notification as they would outside a lab context, without intervention from the experimenter. This is the crucial point where it's seen if participants are able to successfully use the notification presented. Do they correctly remediate the problem? Do they use multi-factor verification if prompted? The experimental computer should have its screen recorded for later analysis and coding of interaction.

Finally, participants should be debriefed with the **post-task survey**. This survey should get into more depth about a participant's perception of the notification, including trustworthiness and their perceived "correct" course of action. Researchers may find it useful to include a free text or interview component here.

A.4 Future work

There is still much work to be done beyond this initial validation study. We see three main paths forward: more detailed design work for the security notification context, further validation of concepts like multi-factor verification, and broader research into appropriate framing and presentation of notifications. These avenues for future work are detailed as part of the larger future work discussion in the Conclusion (Chapter 6).

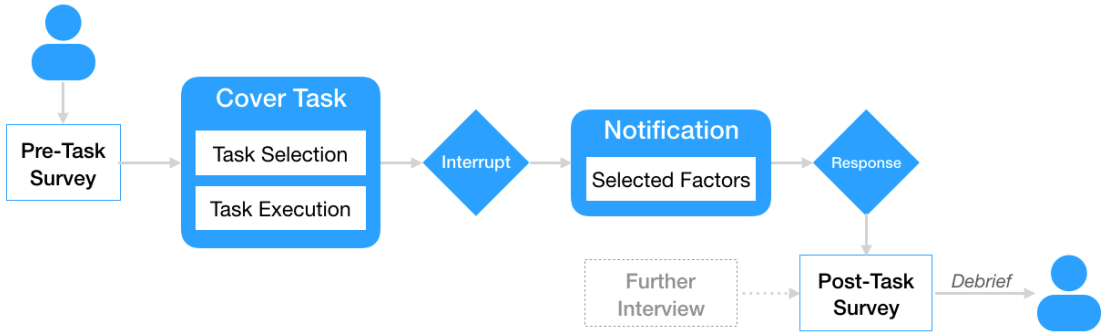

Figure A-2: Participant flow for a generalized user study on H1, H2, or H3.

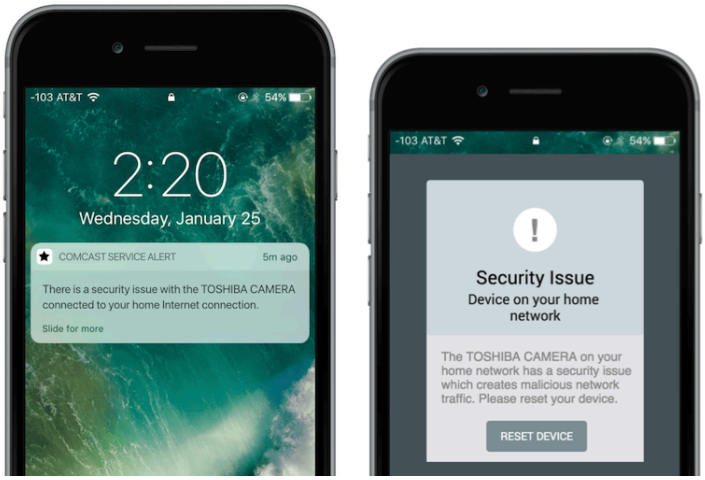

Figure A-3: Potential notification event embedded in a mobile interface

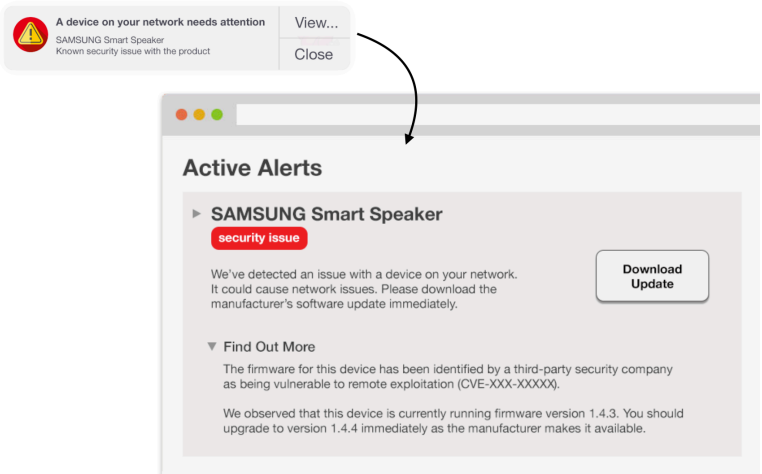

Figure A-4: Potential notification event embedded in a desktop task context.

THIS PAGE INTENTIONALLY LEFT BLANK

Bibliography

[1] ACMA database keeps finger on Australias malware pulse. URL: https://www.cso.com.au/article/462419/acma_database_keeps_finger_australia_malware_pulse/ (visited on 11/15/2017).

[2] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, and M. Kallitsis. Understanding the mirai botnet. In *USENIX Security Symposium*, Pages 1092–1110, 2017.

[3] H. Asghari. Botnet Mitigation and the Role of ISPs: A quantitative study into the role and incentives of Internet Service Providers in combating botnet propagation and activity. en, 2010. URL: http://resolver.tudelft.nl/uuid:db5eac04-61f9-4e1d-8f6e-5cdf3613bf42 (visited on 12/17/2018).

[4] L. Bauer, C. Bravo-Lillo, L. Cranor, and E. Fragkaki. Warning design guidelines. Technical report CMU-CyLab-13-002, Carnegie Mellon University CyLab, 2013.

[5] S. Bauer. Net.info: a provider to subscriber secure communication channel, 2015. URL: http://aqualab.cs.northwestern.edu/component/attachments/download/604.

[6] K. Bicakci, E. Altuncu, M. S. Sahkulubey, H. E. Kiziloz, and Y. Uzunay. How Safe Is Safety Number? A User Study on SIGNALs Fingerprint and Safety Number Methods for Public Key Verification. en. In L. Chen, M. Manulis, and S. Schneider, editors, *Information Security*, Lecture Notes in Computer Science, Pages 85–98. Springer International Publishing, 2018. ISBN: 978-3-319-99136-8.

[7] A.-R. Blais and E. U. Weber. A Domain-Specific Risk-Taking (DOSPERT) Scale for Adult Populations. en. SSRN Scholarly Paper ID 1301089, Social Science Research Network, Rochester, NY, July 2006. URL: https://papers.ssrn.com/abstract=1301089 (visited on 12/19/2018).

[8] C. Bravo-Lillo, L. F. Cranor, J. Downs, and S. Komanduri. Bridging the Gap in Computer Security Warnings: A Mental Model Approach. *IEEE Security Privacy*, 9(2):18–26, Mar. 2011. ISSN: 1540-7993. DOI: 10.1109/MSP.2010.198.

[9] C. I. Canfield, B. Fischhoff, and A. Davis. Quantifying Phishing Susceptibility for Detection and Behavior Decisions. en. *Human Factors*, 58(8):1158–1172, Dec. 2016. ISSN: 0018-7208.

DOI: 10.1177/0018720816665025. URL: https://doi.org/10.1177/0018720816665025 (visited on 01/06/2019).

[10] B. S. D. Carberry and 2. Mar 09. Industry calls for more cyber threat context from DHS -. en. URL: https://fcw.com/articles/2017/03/09/info-sharing-context-industry.aspx (visited on 12/17/2018).

[11] O. Çetin, C. Gañán, L. Altena, S. Tajalizadehkhoob, and M. v. Eeten. Let Me Out! Evaluating the Effectiveness of Quarantining Compromised Users in Walled Gardens. en. In Pages 251–263, 2018. URL: https://www.usenix.org/conference/soups2018/presentation/cetin (visited on 12/15/2018).

[12] O. Cetin, C. Ganán, M. Korczynski, and M. v. Eeten. Make notifications great again: learning how to notify in the age of large-scale vulnerability scanning. In *Workshop on the Economy of Information Security*, 2017.

[13] C. Chung, A. Kasyanov, J. Livingood, N. Mody, and B. Van Lieu. Comcast's Web Notification System Design. en. Technical report RFC 6108, 2011. URL: https://tools.ietf.org/html/rfc6108 (visited on 12/19/2018).

[14] J. Clark and P. C. v. Oorschot. SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements. In *Security and Privacy (SP), 2013 IEEE Symposium on*, Pages 511–525. IEEE, 2013.

[15] Cloudflare. What is a ddos botnet, 2018. URL: https://www.cloudflare.com/learning/ddos/what-is-a-ddos-botnet/.

[16] Combating Spam: Policy, Technical and Industry Approaches. en-US, 2012. URL: https://www.internetsociety.org/resources/doc/2012/combating-spam-policy-technical-and-industry-approaches/ (visited on 01/04/2019).

[17] Cyber Clean Center. The Fight Against the Threat from Botnets: Report on the Activities of the Cyber Clean Center. Technical report, 2011. URL: https://www.telecom-isac.jp/ccc/en_report/Report_on_the_activities_of_the_Cyber_Clean_Center.pdf (visited on 06/07/2017).

[18] B. Dean. Strict Product Liability and the Internet of Things. en. Technical report, Center for Democracy and Technology, Apr. 2018. URL: https://cdt.org/insight/report-strict-product-liability-and-the-internet-of-things/ (visited on 01/16/2019).

[19] Department of Broadband, Communications and the Digital Economy. Review of the Internet Service Providers voluntary code of practice for industry self-regulation in the area of cybersecurity. Technical report, Australian Department of Communications and the Arts, 2013. URL: https://www.communications.gov.au/sites/g/files/net301/f/icode-Review-Report_.pdf (visited on 06/08/2017).

[20] Departments of Commerce and Homeland Security. Enhancing the Resilience of the Internet and Communications Ecosystem Against Botnets and Other Automated, Distributed Threats. Technical report, May 2018. URL: https://www.commerce.gov/sites/default/files/media/files/2018/eo_13800_botnet_report_-_finalv2.pdf.

[21] R. Dhamija and J. D. Tygar. The Battle Against Phishing: Dynamic Security Skins. In *Proceedings of the 2005 Symposium on Usable Privacy and Security*, SOUPS '05, Pages 77–88, New York, NY, USA. ACM, 2005. ISBN: 978-1-59593-178-8. DOI: 10.1145/1073001.1073009. URL: http://doi.acm.org/10.1145/1073001.1073009 (visited on 02/22/2018).

[22] R. Dhamija, J. D. Tygar, and M. Hearst. Why Phishing Works. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, Pages 581–590, New York, NY, USA. ACM, 2006. ISBN: 978-1-59593-372-0. DOI: 10.1145/1124772.1124861. URL: http://doi.acm.org/10.1145/1124772.1124861 (visited on 01/02/2019).

[23] M. v. Eeten, Q. Lone, G. Moura, H. Asghari, and M. Korczyski. Evaluating the Impact of AbuseHUB on Botnet Mitigation. *arXiv:1612.03101 [cs]*, Dec. 2016. URL: http://arxiv.org/abs/1612.03101.

[24] S. Egelman, L. F. Cranor, and J. Hong. You'Ve Been Warned: An Empirical Study of the Effectiveness of Web Browser Phishing Warnings. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, Pages 1065–1074, New York, NY, USA. ACM, 2008. ISBN: 978-1-60558-011-1. DOI: 10.1145/1357054.1357219. URL: http://doi.acm.org/10.1145/1357054.1357219 (visited on 01/06/2019).

[25] S. Egelman and E. Peer. Scaling the security wall: Developing a security behavior intentions scale (sebis). In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, Pages 2873–2882. ACM, 2015.

[26] M. Fagan and M. M. H. Khan. Why Do They Do What They Do?: A Study of What Motivates Users to (Not) Follow Computer Security Advice. en. In Pages 59–75, 2016. URL: https://www.usenix.org/conference/soups2016/technical-sessions/presentation/fagan (visited on 12/16/2018).

[27] FCC CSIRC WG 7. U.S. Anti-Bot Code of Conduct (ABC) for Internet Services Providers (ISPs). Technical report, June 2017. URL: https://www.m3aawg.org/system/files/20120322_WG7_Final_Report_for_CSRIC_III_5_0.pdf (visited on 06/06/2017).

[28] FCC CSIRC WG 7. U.S. Anti-Bot Code of Conduct (ABC) for Internet Services Providers (ISPs): Barriers and Metric Considerations. Technical report, June 2017. URL: https://transition.fcc.gov/bureaus/pshs/advisory/csric3/CSRIC_III_WG7_Report_March_%202013.pdf (visited on 06/06/2017).

[29] Federal Trade Commission. Credit Card Interest Rate Reduction Scams. en, Feb. 2011. URL: https://www.consumer.ftc.gov/articles/0131-credit-card-interest-rate-reduction-scams (visited on 01/02/2019).

[30] A. P. Felt, A. Ainslie, R. W. Reeder, S. Consolvo, S. Thyagaraja, A. Bettes, H. Harris, and J. Grimes. Improving SSL Warnings: Comprehension and Adherence. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, Pages 2893–2902, New York, NY, USA. ACM, 2015. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702442. URL: http://doi.acm.org/10.1145/2702123.2702442 (visited on 11/02/2017).

[31] GlobalPlatform, Inc. Introduction to Trusted Execution Environments. Technical report, 2018. URL: https://globalplatform.org/wp-content/uploads/2018/05/Introduction-to-Trusted-Execution-Environment-15May2018.pdf.

[32] D. Goodin. Tech-support scammers revive bug that sends Chrome users into a panic. en-us, July 2018. URL: https://arstechnica.com/information-technology/2018/07/tech-support-scammers-revive-bug-that-sends-chrome-users-into-a-panic/ (visited on 01/01/2019).

[33] T. Grenman. AutoreporterKeeping the Finnish Network Space Secure. Technical report, CERT-FI, 2009.

[34] R. Kang, L. Dabbish, N. Fruchter, and S. Kiesler. My Data Just Goes Everywhere: User Mental Models of the Internet and Implications for Privacy and Security. *SOUPS 2015*:39, 2015.

[35] R. Kartch. Distributed Denial of Service Attacks: Four Best Practices for Prevention and Response. en-us, Nov. 2016. URL: https%3A%2F%2Finsights.sei.cmu.edu%2Fsei_blog%2F2016%2F11%2Fdistributed-denial-of-service-attacks-four-best-practices-for-prevention-and-response.html (visited on 01/02/2019).

[36] Knowledge-Based Verification. URL: https://csrc.nist.gov/Glossary/Term/Knowledge_Based-Verification.

[37] Korean Internet Security Agency. Safe Internet, Happy Future! Technical report, 2015. URL: https://www.sbs.ox.ac.uk/cybersecurity-capacity/content/information-security-korea-safe-internet-happy-future.

[38] B. Krebs. Study: Attack on KrebsOnSecurity Cost IoT Device Owners $323k Krebs on Security. en-US, 2018. URL: https://krebsonsecurity.com/2018/05/study-attack-on-krebsonsecurity-cost-iot-device-owners-323k/ (visited on 01/05/2019).

[39] F. Li, Z. Durumeric, J. Czyz, M. Karami, M. Bailey, D. McCoy, S. Savage, and V. Paxson. You've Got Vulnerability: Exploring Effective Vulnerability Notifications. In *USENIX Security Symposium*, Pages 1033–1050, 2016.

[40] J. Livingood, N. Mody, and M. O'Reirdan. Recommendations for the Remediation of Bots in ISP Networks. en. Technical report RFC 6561, 2012. URL: https://tools.ietf.org/html/rfc6561 (visited on 01/05/2019).

[41] G. Macri. Only One Company is Using DHS's Automated Cyber Threat Sharing Portal, Sept. 2016. URL: http://www.insidesources.com/only-one-company-using-dhss-automated-cyber-threat-sharing-portal/ (visited on 01/02/2018).

[42] B. Matthews. Inquiry regarding AISI research reports, Nov. 2017.

[43] B. Matthews. The Australian Internet Security Initiative. Technical report 2010/TEL41/SPSG/WKSP1/005, APEC, 2010.

[44] Models To Advance Voluntary Corporate Notification to Consumers Regarding the Illicit Use of Computer Equipment by Botnets and Related Malware, Sept. 2011. URL: https://www.federalregister.gov/documents/2011/09/21/2011-24180/models-to-advance-

voluntary-corporate-notification-to-consumers-regarding-the-illicit-use-of (visited on 01/05/2019).

[45] M. G. Morgan, B. Fischhoff, A. Bostrom, and C. J. Atman. *Risk Communication: A Mental Models Approach*. en. Cambridge University Press, 2002. ISBN: 978-0-521-80223-9. Google-Books-ID: ieXbkmYf3mAC.

[46] B. Nadel. F-Secure Sense Review. en, Nov. 2017. URL: https://www.tomsguide.com/us/f-secure-sense,review-4801.html (visited on 12/19/2018).

[47] J. Nielsen. Progressive Disclosure. en, 2006. URL: https://www.nngroup.com/articles/progressive-disclosure/ (visited on 01/03/2019).

[48] Norton Core Router | Secure WiFi Router. en. URL: https://us.norton.com/core (visited on 12/19/2018).

[49] Presidential Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure, May 2017. URL: https://www.whitehouse.gov/the-press-office/2017/05/11/presidential-executive-order-strengthening-cybersecurity-federal (visited on 06/28/2017).

[50] C. Rebaza, C. Shoichet, and D. Williams. Did tear gas prompt London City Airport evacuation? URL: https://edition.cnn.com/2016/10/21/europe/london-city-airport-evacuated/index.html (visited on 12/26/2018).

[51] E. M. Redmiles, A. R. Malone, and M. L. Mazurek. I Think They're Trying to Tell Me Something: Advice Sources and Selection for Digital Security. In *2016 IEEE Symposium on Security and Privacy (SP)*, Pages 272–288, May 2016. DOI: 10.1109/SP.2016.24.

[52] R. W. Reeder, A. P. Felt, S. Consolvo, N. Malkin, C. Thompson, and S. Egelman. An Experience Sampling Study of User Reactions to Browser Warnings in the Field. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, 512:1–512:13, New York, NY, USA. ACM, 2018. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3174086. URL: http://doi.acm.org/10.1145/3173574.3174086 (visited on 12/19/2018).

[53] H. T. Reis and C. M. Judd. *Handbook of Research Methods in Social and Personality Psychology*. en. Cambridge University Press, Mar. 2000. ISBN: 978-0-521-55903-4. Google-Books-ID: j7aawGLbtEoC.

[54] S. Ruoti, J. Andersen, D. Zappala, and K. Seamons. Why Johnny still, still can't encrypt: Evaluating the usability of a modern PGP client. *arXiv preprint arXiv:1510.08555*, 2015.

[55] C. Seaman. Akamai threat advisory: Mirai botnet. Technical report, Technical Report. Akamai Technologies, 2016.

[56] Servicio AntiBotnet | Oficina de Seguridad del Internauta. URL: https://www.osi.es/es/servicio-antibotnet (visited on 01/05/2019).

[57] S. Sheng, B. Magnien, P. Kumaraguru, A. Acquisti, L. F. Cranor, J. Hong, and E. Nunge. Anti-Phishing Phil: The Design and Evaluation of a Game That Teaches People Not to Fall for Phish. In *Proceedings of the 3rd Symposium on Usable Privacy and Security*, SOUPS '07, Pages 88–99, New York, NY, USA. ACM, 2007. ISBN: 978-1-59593-801-5. DOI: 10.1145/

1280680.1280692. URL: http://doi.acm.org/10.1145/1280680.1280692 (visited on 01/06/2019).

[58] S. Shoji. Magnitude 6.6 Quake Causes Strong Shaking in Western Japan. en, Oct. 2016. URL: https://www.bloomberg.com/news/articles/2016-10-21/magnitude-6-6-quake-hits-western-japan-causing-strong-shaking (visited on 12/27/2018).

[59] A. Shostack. *Threat modeling: Designing for security*. John Wiley & Sons, 2014.

[60] Signal >> Blog >> Safety number updates. URL: https://signal.org/blog/safety-number-updates/ (visited on 01/06/2019).

[61] SISSDEN | About, 2018. URL: https://sissden.eu/ (visited on 01/05/2019).

[62] Spectrum ISP botnet warning. en-us. URL: https://www.reddit.com/r/techsupport/comments/8numwk/spectrum_isp_botnet_warning/ (visited on 12/17/2018).

[63] D. W. Stewart and I. M. Martin. Intended and Unintended Consequences of Warning Messages: A Review and Synthesis of Empirical Research. *Journal of Public Policy & Marketing*, 13(1):1–19, 1994. ISSN: 0743-9156. URL: https://www.jstor.org/stable/30000168 (visited on 12/17/2018).

[64] B. Stock, G. Pellegrino, F. Li, M. Backes, and C. Rossow. Didnt You Hear Me? Towards More Successful Web Vulnerability Notifications. en. In Feb. 2018. URL: https://publications.cispa.saarland/1190/ (visited on 12/15/2018).

[65] B. Stock, G. Pellegrino, C. Rossow, M. Johns, and M. Backes. Hey, You Have a Problem: On the Feasibility of Large-Scale Web Vulnerability Notification. In *USENIX Security Symposium*, Pages 1015–1032, 2016.

[66] The Australian Communications and Media Authority. The AISI interviews with industry. English, Web pages, Sept. 2015. URL: http://www.acma.gov.au/Industry/Internet/e-Security/Australian-Internet-Security-Initiative/the-aisi-interviews-with-industry (visited on 06/07/2017).

[67] E. Vaziripour, J. Wu, M. O'Neill, D. Metro, J. Cockrell, T. Moffett, J. Whitehead, N. Bonner, K. Seamons, and D. Zappala. Action Needed! Helping Users Find and Complete the Authentication Ceremony in Signal. en. In Pages 47–62, 2018. URL: https://www.usenix.org/conference/soups2018/presentation/vaziripour (visited on 01/06/2019).

[68] R. Wash, E. Rader, K. Vaniea, and M. Rizor. Out of the loop: How automated software updates cause unintended security consequences. In *Symposium on Usable Privacy and Security (SOUPS)*, Pages 89–104, 2014.

[69] J. Weidman and J. Grossklags. I Like It, but I Hate It: Employee Perceptions Towards an Institutional Transition to BYOD Second-Factor Authentication. In *Proceedings of the 33rd Annual Computer Security Applications Conference*, ACSAC 2017, Pages 212–224, New York, NY, USA. ACM, 2017. ISBN: 978-1-4503-5345-8. DOI: 10.1145/3134600.3134629. URL: http://doi.acm.org/10.1145/3134600.3134629 (visited on 12/17/2018).

[70] M. Weirich, P. Meyer, T. Kraft, S. Bleidner, T. King, M. Simonis, G. RoSSrucker, T. Berchem, and A. Zeh. D1.2.2 Specification of Tool Group Centralised Data Clearing House. Technical

report D1.2.2, 2015. URL: https://acdc-project.eu/wp-content/uploads/2016/05/D1.2.2_ACDC_Centralized_Data_Clearing_House-s.pdf (visited on 01/05/2019).

[71] A. Whitten and J. D. Tygar. Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. In *USENIX Security Symposium*, volume 348, 1999.

[72] Xfinity. Xfinity Bot Notifications Policy. en, Apr. 2012. URL: https://www.xfinity.com/support/articles/constant-guard-bot-notifications-policy (visited on 12/17/2018).