Formally Reasoning about the Cost and Efficacy of Securing the Email Infrastructure

Patrick Speicher[*], **Marcel Steinmetz**[*], **Robert Künnemann**[*], **Milivoj Simeonovski**[*], **Giancarlo Pellegrino**[†], **Jörg Hoffmann**[*], and **Michael Backes**[‡]

[*]*CISPA, Saarland University, Saarland Informatics Campus*

[†]*CISPA, Saarland University, Stanford University*

[‡]*CISPA Helmholtz Center i.G., Saarland Informatics Campus*

Email: {first name}.{last name}@cispa.saarland, {gpellegrino, backes}@cispa.saarland, hoffmann@cs.uni-saarland.de

***Abstract*—Security in the Internet has historically been added post-hoc, leaving services like email, which, after all, is used by 3.7 billion users, vulnerable to large-scale surveillance. For email alone, there is a multitude of proposals to mitigate known vulnerabilities, ranging from the introduction of completely new protocols to modifications of the communication paths used by big providers. Deciding which measures to deploy requires a deep understanding of the induced benefits, the cost and the resulting effects.**

This paper proposes the first automated methodology for making formal deployment assessments. Our planning algorithm analyses the impact and cost-efficiency of different known mitigation strategies against an attacker in a formal threat model. This novel formalisation of an infrastructure attacker includes routing, name resolution and application level weaknesses. We apply the methodology to a large-scale scan of the Internet, and assess how protocols like IPsec, DNSSEC, DANE, SMTP STS, SMTP over TLS and other mitigation techniques like server relocation can be combined to improve the confidentiality of email users in 45 combinations of attacker and defender countries and nine cost scenarios. This is the first deployment analysis for mitigation techniques at this scale.

1. Introduction

The Internet infrastructure relies on the correct functioning of the basic underlying protocols for routing and name resolution, which, historically, were designed for functionality, not for security. Over the past decades, the security research community has put a lot of effort into strengthening the security of these protocols by applying cryptographic means on various layers of communication (e.g., IPsec, DNSSEC, TLS). But only a small fraction of them have been deployed and have a real-world impact. Is inertia the only reason that this is the case? For email alone, there are a multitude of proposals to mitigate known vulnerabilities, ranging from the introduction of completely new protocols to modifications of the communication paths used by big providers [1], [2], [3], [4], [5], [6], [7]. At the very core, every meaningful decision about which measure to deploy calls for a deep and rigorous understanding of the induced benefits, the cost and the resulting effects.

In this work, we provide a thorough and automated methodology for making formal deployment assessments for various layers of the email infrastructure. We propose such a methodology in the form of a *mitigation analysis*. Our methodology models and analyses the impact of different mitigation strategies against an attacker as a game with a single exchange of action and counter-action, similar to a Stackelberg game. Such a mitigation analysis was recently proposed for corporate networks, where the number of hosts is in the hundreds [8]. Here, we adapt the model and analysis to Internet infrastructure attack mitigation, over a dataset spanning more than 6 million domains, IPs, autonomous systems and DNS-zones. Given the scale of the data set, the computational challenge is tremendous, as the best attacker strategy needs to be considered *for every combination of mitigation choices on the part of the defender.* Tackling this complexity is where security meets AI: following Speicher et al. [8], we employ AI *planning* methods for effective search in mitigation-choice space, geared at finding good choices quickly and pruning later choices against the bounds thus identified. Finally, the outcome of the mitigation analysis is the *Pareto frontier* of mitigation strategies, i.e., the optimal choice of mitigation efforts per budget. In practice, this provides a function from a given budget to the set of most effective mitigation strategies. Such an assessment is useful (*a*) for standardisation bodies to estimate the potential improvement of the adoption of a new security standard, (*b*) for government bodies to evaluate strategies for digital self-reliance, (*c*) for protocol designers to guide and evaluate their decisions w.r.t. the current infrastructure, and (*d*) for providers to make investment decisions on their infrastructure.

We demonstrate our methodology at scale in a case study where we take the high-level perspective of a government body seeking to secure its Internet infrastructure. In particular, we focus on the case of email, which is used by 3.7 billion users worldwide [9]. Given publicly available information and ballpark cost estimates for mitigations, we assess the cost and impact of mitigations in nine cost scenarios and 45 combinations of defender and attacker countries.

Our results suggest a nuanced view. There is no single cure for all, viable mitigation strategies differ from country to country and scenario to scenario.

To summarize, we make the following contributions:

- We present a comprehensive deployment cost estimation methodology, reasoning about the cost versus the efficacy of different mitigation strategies (Section 2).
- We present a novel, formal threat model covering attacks on the routing level, name resolution and email communication (Section 3).
- We formalize the effects and estimate the cost of proposals like IPsec, DNSSEC, DANE, SMTP STS and SMTP over TLS (SMTPS), yielding a defender model composed of mitigation strategies for minimizing the attacker's objective (Section 4).
- We apply our methodology on the German email infrastructure in face of large-scale email sniffing and discuss different cost scenarios (Section 6.2), and 45 combinations of attacker and defender countries (Section 6.3).
- Here, we identify viable mitigation strategies in different scenarios, as well as strategies that are only rarely worth the effort, while also highlighting deployment issues of existing approaches and their limits in face of foreign infrastructure.

Limitations. Our analysis relies on (i) a formal threat model and (ii) a formal defender model. Our threat model assumes the protocols covered here to work as intended; even if this is the case, as of now there is no formal justification for it to be sound and complete, e.g., w.r.t. a Dolev-Yao attacker and a correct implementation of the respective protocols. As we compare attacks by their effectiveness, both soundness and completeness are required. Our defender-model depends on a realistic cost assessment. Given the scope of this paper, we can provide only a rudimentary analysis that sacrifices precision for uniformity and clarity. We made an effort to provide sources for our cost assessment where possible, but point out that (a) cost vary from company to company and often depend on company secrets (b) our cost model includes only direct monetary cost (c) we do not quantify the margin of error of our results.

2. Planning

Our mitigation analysis is based on *Automated Planning*, one of the oldest sub-areas of AI (see [10] for a comprehensive introduction). Given a high-level description of the relevant world properties consisting of *state propositions*, *initial state*, a *goal* specification, and a set of *actions* that can be used to alter the state of the world, the basic principle behind automated planning is to find a combination of actions that achieve the goal condition when applied in the initial state. Different forms of planning are distinguished by their modeling assumptions, which determine the complexity class that typical questions, e.g., existence of a plan, fall into. In the following, we will focus entirely on *classical planning*, where it is assumed that all actions have deterministic effects, and that the initial state is completely known, up front – both being reasonable assumptions in our model.

```
 1) Reward = 0.217100, Cost = $0
    ∅

 2) Reward = 0.161200, Cost = $366 342
    nDNSSEC(mail.ru)
    .
    .
    .
15) Reward = 0.010400, Cost = $168 372 779
    nDNSSEC(rambler.ru), nDNSSEC(mail.ru), nDNSSEC(yandex.ru),
    nReloc(gmail.com), nReloc(yandex.ru), nReloc(mail.ru)

16) Reward = 0.000000, Cost = $186 404 888
    nDNSSEC(rambler.ru), nDNSSEC(mail.ru), nDNSSEC(yandex.ru),
    nReloc(gmail.com), nRFC7819(yandex.ru), nRFC7819(rambler.ru),
    nRFC7819(mail.ru)
```

Figure 1: Excerpt of the Pareto frontier from USA vs. Russia.

Planning has been used in a range of application as diverse as the control of modular printers [11], natural language sentence generation [12], greenhouse logistics [13] and, in particular, network security penetration testing [14], [15], [16]. This latter branch of research – network attack planning as a tool for automated security testing – has been called *simulated pentesting*. We adopt this approach here to our context.

2.1. Mitigation analysis

Mitigation analysis through planning has been proposed recently in the context of network penetration testing [8] and can be seen as a two-fold planning task, in which a defender plans for mitigation strategies that impose a limit on the worst any attacker can do.

Formally, the state of the world and the state of an attack are described through a finite set of propositions. A state is a truth value assignment to these propositions. Concrete attacks depend on the initial world state and are determined from a finite set of *attacker actions*. Associated with an attack is the *attacker reward* $\mathcal{R}$, which is used as an indicator of the severity of the attack. For example, as shown in Section 3.1, in our model the reward of an attack corresponds to the number of connections that could be compromised via the attack. To prevent attacks and thus to lower the attacker reward, the defender can change the world state through the application of *defender actions*.

In logical terms, an action is given by a *precondition* pre, a boolean formula over proposition literals and a *postcondition* $post$, a conjunction over proposition literals, and is written as

$$\frac{pre}{post}.$$

An action may only be applied in states that satisfy the action's precondition. In the state resulting from this application, all propositions with positive occurrence in $post$ are made true, and vice versa, all propositions with negative occurrence in $post$ are made false. Additionally, every defender action is associated with a positive real *cost* value.

In the concrete attacker-planning model that we derive from our dataset (detailed in Section 3), we employ *monotonic* attacker actions, with positive preconditions and

postconditions only (no negated literals). This corresponds to a wide-spread assumption in automated attack analysis, specifically attack analysis via *attack graphs* (e.g. [17], [18]): during the course of a given attack, the attacker incrementally gains new assets, but never loses any assets (we will discuss the attack graph literature in more detail in Section 7). In other words, in our mitigation analysis, the defender gets to move first, and the subsequent move of the attacker is carried out without additional interference on the part of the defense. Planning with monotonic actions is polynomial-time while planning with general action models is **PSPACE**-complete [19].

We compare different *mitigation strategies*, i.e., sequences σ of defender actions, in terms of their cost $c(\sigma)$: the sum of the cost of the defender actions in the sequence; and the impact on attacks $\mathcal{R}(\sigma)$: the attacker reward in the state resulting from the application of σ. We say that a mitigation strategy *dominates*, 'is better than', another mitigation strategy if its cost is strictly smaller while the attacker reward is not larger, or if the maximal attacker reward is strictly smaller while the cost is not larger, i.e., either $\mathcal{R}(\sigma) \leq \mathcal{R}(\sigma')$ and $c(\sigma) < c(\sigma')$, or $\mathcal{R}(\sigma) < \mathcal{R}(\sigma')$ and $c(\sigma) \leq c(\sigma')$. Our analysis yields the *Pareto frontier* $\mathcal{P}$ of mitigation strategies: the set of mitigation strategies which are not dominated by any other mitigation strategy. Figure 1 gives an example of a Pareto frontier output by our analysis tool, the details of which will become clear in Section 6.

2.2. Planning algorithm

To compute the Pareto frontier, we extend the algorithm proposed for network penetration testing [8] to our setting. In its basic principle, the algorithm tests and compares all mitigation strategies against each other in order to find the non-dominated ones. However, naïvely comparing all possible mitigation strategies is infeasible even for small models. The overall number of mitigation strategies that have to be compared is exponential in the number of defender actions. To scale to larger and hence more interesting models from a practical perspective, Speicher et al. [8] proposed several pruning techniques to skip mitigation options which are irrelevant or inefficient considering the current network state and the mitigations considered so far. In doing so, it was possible to analyze networks of up to about 1 000 hosts in their evaluation. We adopt these techniques in our work, but even with these optimizations, analyzing data at Internet scale would be far beyond the scope of the algorithm. To make it feasible to run this mitigation analysis approach on our model, we identify parts that are relevant for neither the attacker nor for the defender, and that can thus be removed prior to the analysis. The identification of irrelevant parts here is based on the concept of *property graphs*.

2.3. Property Graph

A basic input to our planning model machinery is the property graph, a labeled graph introduced by Simeonovski et al. [20]. Nodes in this graph represent IP addresses, domain names, Autonomous Systems (AS, a group of routers whose addresses and routing polices are under common administrative control) and countries; edges represent relationships between them.

node/edge	description
IP	IP address.
Dom	domain name.
AS	IANA number assigned to the AS.
Cntry	country code.
ORIG	*AS* where an lhs originates from.
LOC	*country* where lhs (IP $\cup$ Dom $\cup$ AS) is located.
DNS	resolving lhs domain requires query to rhs.
RSLVR	lhs uses resolver on rhs for name resolution.
A	DNS record mapping Dom to IP.
MX	DNS record mapping Dom for email delivery.
RTE(AS_t)	AS-level route via AS_t between two *ASes*.

TABLE 1: Labels of nodes and relationships

We extend the graph formalism from prior work with AS-level routes and an indication of DNS resolvers used. The AS-level routes indicate the ASes that a packet traverses when transmitted from source to destination. These are directed, as routes are not always symmetrical. The resolver, or local DNS resolver, is the last recursive name server that performs iterative domain resolution. We will explain how the DNS resolver is identified and the routing data collected in Section 5.

Figure 2 presents a snippet of the property graph. Nodes in the set Dom indicate domain names, e.g., `gmail.com` with a DNS record A mapping the IP addresses (nodes in the set IP), e.g., 216.58.207.37, which in turn belongs to an AS (AS15169) and is geolocated in the US. A directed first-order edge RTE(AS_t) (AS_t is a property of this edge) indicates that there is *some* route from the source AS to the destination AS which traverses AS_t.

For instance, in our example the edge labels show that AS_1 to AS_n might be traversed if a user using `gmail.com` sends an email to `t-online.de`. The set of labels $\{AS_1, \ldots, AS_n\}$ is the union of ASes that appear in some route, but not a route itself, as we only need data at this granularity. The graph does not include peering agreements between ASes. Given these agreements and each AS'es routing policy, these routing edges can be computed; we choose to probe routes used instead. Table 1 summarizes the types of nodes and edges which we consider in our model. The threat model described in the next section is formulated in terms of rules with predicates of form $a \xrightarrow{B} c$ or $a \in S$ for the existence of labeled edges and node types in the property graph.

2.4. Precomputation

Our analysis determines the relevant part of the graph by combining a tainting similar to the approach proposed by Simeonovski et al. [20] with a reversed tainting starting from the mail providers. Only nodes tainted from both directions need to be considered, significantly reducing the size of

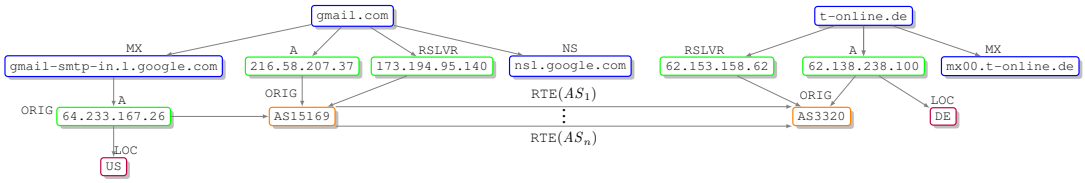

Figure 2: Snippet of the property graph

the graph relevant for the analysis. More precisely: given the full graph, all attacker rules are applied until a fixed point is reached. These are monotonous, i.e., the order in which they are applied is unimportant (cf. Section 3). All nodes and edges appearing in any instantiation of a rule that was applied are 'forward tainted'. For any provider that was forward tainted, we proceed in the opposite direction, using only actions that were applied in the forward tainting. Interpreting them backward, we again apply them until a fixed point is reached. Nodes and edges appearing in these actions are now forward and backward tainted and define the relevant sub graph of the problem.

3. Threat Model

Since the Snowden revelations have unfolded, governments are aware of and react to large-scale spying programs targeting 'nearly everything a typical user does on the Internet' via deep packet inspection and cooperation of domestic companies [21]. Hence our threat model considers an infrastructure attacker who is able to use routing and DNS spoofing to mount attacks which are not specific to implementation vulnerabilities or protocol weaknesses of SMTP, but concern the underlying assumptions on routing and domain resolutions that the security of protocols without endpoint verification, such as SMTP, rely on. Furthermore, the attacking country has the means to compromise servers located in her jurisdiction, e.g., using gag orders or legislature, and to observe and intercept packets routed via ASes under her jurisdiction. The goal of the attacker is large-scale surveillance, hence the attacker is active, but restricted to easy-too-mount attacks which scale well and avoid global exposure. For this reason, we do not consider off-path attacks on DNS or false BGP announcements.

In contrast to formal protocol verification, our threat model considers the combination of known attack vectors w.r.t. routing, name resolution, and email communication in a more abstract, attack-centric view. The attacker is not in full control over the network but can inject packets at certain routes.

3.1. Attacker reward

The goal of the attacker is to observe as much of the defender country's email communication as possible. Instead of counting the number of mail domains the adversary can gain control of, we estimate the impact in terms of users affected based on market share data and the total number of Internet users in the defending country. Not only does this provide us with a meaningful impact metric, but it facilitates the analysis, as in most countries email is a market with very few players. We represent those as a set of domains $\mathsf{Provider} \subseteq \mathsf{Dom}$ and consider only providers for which market share data is available. The ability to intercept emails from provider A to provider B is represented by a predicate $\mathsf{unconf}(A, B)$. The attacker optimises the number distinct tuples A, B, weighted by their respective share of communication $\omega(A, B)$, which is the product of the market share of A and B: $\mathcal{R} = \max \sum_{d,e \in \mathsf{Provider\ s.t.\ unconf}(d,e)} \omega(d, e)$[1].

3.2. Attacker actions

We formalize the threat model as a set of attack rules, which, when instantiated with all domains, countries, ASes, IPs, etc. give us a large but finite set of attacker actions. All attacks we present here are well known, the contribution of this section is limited to their formalisation.

We distinguish between domains, IPs or ASes that initially belong to the adversary, domains that do not belong to the adversary, but can be resolved to adversarial IPs, and MXes which can be resolved to adversarial domains. This corresponds to direct (initial) compromise in the real-world, compromise on the DNS layer, and compromise on the application layer, i.e., mail (which relies on DNS). Similarly, we distinguish between compromise of communication on the routing, DNS, and application layer. Table 2 gives an overview of the predicates we use in our modelling.

3.2.1. Initially Compromised Nodes. The attacker starts with a set of *nodes* considered compromised initially, namely ASes, IPs and domains located in the attacking country.

$$\frac{x \in \mathsf{AS} \cup \mathsf{IP} \cup \mathsf{Dom} \quad cn \in \mathsf{Cntry} \quad x \xrightarrow{\mathrm{LOC}} n \quad \mathsf{C}(cn)}{\mathsf{C}(x)}$$

An IP is also considered compromised if it belongs to an attacking country or a compromised AS. If a domain resolves to such an IP, this domain is considered compromised, too.

$$\frac{i \in \mathsf{IP} \quad a \in \mathsf{AS} \quad i \xrightarrow{\mathrm{ORIG}} a \quad \mathsf{C}(a)}{\mathsf{C}(i)} \qquad \frac{d \in \mathsf{Dom} \quad i \in \mathsf{IP} \quad d \xrightarrow{\mathrm{A}} i \quad \mathsf{C}(i)}{\mathsf{C}(d)}$$

Whether or not a server is part of a given jurisdiction is subject to the attacker's legal system and out of the scope of this work; we side-step this question by assuming jurisdiction to equal geographical location. Policy makers

1. A concrete interpretation can be given as follows: if every email user in the defending country sends an email to every other user (technically including himself, but the error is negligible), the rewards equals the percentage of emails the attacker can read.

$\mathsf{C}(x)$	Node $x \in$ Dom $\cup$ IP $\cup$ AS $\cup$ Cntry under adversarial control.
$\mathsf{I}^{\mathsf{DNS}}(d)$	Integrity of name resolution of $d \in$ Dom compromised
$\mathsf{I}^{\mathsf{R}}(d,e)$	Integrity of *some* route from $d \in$ Dom to $e \in$ Dom is compromised.
$\mathsf{I}^{\mathsf{DNS}}(d,e)$	Integrity of name resolution of $e \in$ Dom from perspective of $d \in$ Dom compromised.
$\mathsf{unconf}(d,e)$	email communication going from some user of $d \in$ Provider to some user of $e \in$ Provider is considered unconfidential.
$\mathsf{nDNSSEC}(d)$	$d \in$ Dom does not support DNSSEC.
$\mathsf{nTLS}^{\mathsf{snd}}(d)$	$d \in$ Dom does not enforce strict host validation when sending emails.
$\mathsf{nVPN}(a,b)$	communication between $a,b \in$ AS is not secured with IPsec.
$\mathsf{nDANE}^{\mathsf{rcv}}(d)$	$d \in$ Dom does not support DANE.
$\mathsf{nRFC7819}(d)$	$d \in$ Dom does not validate the receiving server's domain part of the email address according to RFC 7819 [22] when sending emails.

TABLE 2: Integrity and confidentiality impact predicates.

should explicitly state all compromised domains, ASes, and IPs based on an actual legal assessment.

3.2.2. Attacks via Routing. Next, we consider routing attacks at the AS level. Each IP is part of an AS. An IP packet may traverse several AS on the way from source AS to target AS. If *any* of these is compromised, this route is considered compromised.

$$\frac{\begin{array}{c}d,e\in\text{Dom} \quad i,j\in\text{IP} \quad a,b,c\in\text{AS} \quad \mathsf{C}(b) \quad \mathsf{nVPN}(a,c)\\ d\xrightarrow{\text{A}}i \quad e\xrightarrow{\text{A}}j \quad i\xrightarrow{\text{ORIG}}a \quad j\xrightarrow{\text{ORIG}}c \quad a\xrightarrow{\text{RTE}(b)}c\end{array}}{\mathsf{I}^{\mathsf{R}}(d,e)}$$

Routing level mitigations presented in the next section consider securing the entire communication between two ASes a and c using IPsec. The *absence* of this mitigation is represented via a predicate $\mathsf{nVPN}(a,c)$, hence, if this predicate is true, then the integrity of the communication from $d \in$ Dom to $e \in$ Dom must be considered compromised, if there is some compromised AS b on some (possibly multi-hop) route from a to c.

3.2.3. Integrity of domain/MX resolution. A domain's resolver (i.e., the last recursive name server, c.f. Section 5) performs iterative domain resolution whenever it cannot respond with a cached DNS record. Let $d \xrightarrow{\text{DNS}} e$ be given for any name server d that could be queried during resolution of domain e. This would include all root servers, no matter which domain e is chosen. However, we explicitly exclude those, as such an attack would put the DNS infrastructure as a whole into question, and we assume the adversary is sneaky. If there is a compromised domain potentially queried during resolution, the whole resolution is seen as compromised.

$$\frac{d,e\in\text{Dom} \quad d\xrightarrow{\text{DNS}}e \quad \mathsf{C}(e)}{\mathsf{I}^{\mathsf{DNS}}(d)} \qquad (r_{dns-ns})$$

If the resolver itself is already under control of the attacker, then every resolution via this resolver must be seen as compromised.

$$\frac{d,e\in\text{Dom} \quad i\in\text{IP} \quad d\xrightarrow{\text{RSLVR}}i \quad \mathsf{C}(i)}{\mathsf{I}^{\mathsf{DNS}}(d,e)}$$

We also model that the name resolution itself is susceptible to routing attacks, and maintain another predicate, $\mathsf{I}^{\mathsf{DNS}}(d,e)$, formalising a lack of integrity of the domain resolution of e *from the perspective of a domain* d in case that the resolution d performs can be compromised via packet routing. In Section 4, we will consider DNSSEC, which mitigates DNS spoofing via packet injection but not DNS spoofing via compromised authoritative name servers or a compromised resolver[2]. Hence we add the precondition $\mathsf{nDNSSEC}(e)$ for e the domain to be resolved.

$$\frac{\begin{array}{c}d,e,f,r\in\text{Dom} \quad d\xrightarrow{\text{RSLVR}}r \quad \mathsf{nDNSSEC}(e)\\ (e\xrightarrow{\text{DNS}}f\wedge\mathsf{I}^{\mathsf{R}}(r,f))\vee\mathsf{I}^{\mathsf{R}}(d,r)\end{array}}{\mathsf{I}^{\mathsf{DNS}}(d,e)}$$

(Remark: we use a disjunction in the precondition for brevity. This is to be read as two rules, one for each disjunct.)

3.2.4. Confidentiality. The remaining attacker actions translate initial compromise and attacks on the routing and the name resolution to a loss of confidentiality on the protocol level. A peculiarity of the SMTP protocol is that emails are not delivered to the domain indicated by the email, but to the MX entry for this domain, i.e., a domain, which in turn is resolved to an IP. This dependence of application level features from name resolution has undesired effects on certificate validation, as we will see.

If a mail server is initially compromised, all connections to or from it are considered unconfidential immediately.

$$\frac{d,e\in\mathsf{Provider} \quad d\xrightarrow{\text{MX}}d' \quad e\xrightarrow{\text{MX}}e' \quad \mathsf{C}(e')\vee\mathsf{C}(d')}{\mathsf{unconf}(d,e)}$$

If the integrity of the name resolution of an MX is compromised, TLS may preserve the confidentiality of communication by validating the certificate of the recipient. However, this requires the recipient to support TLS (which was the case for all mail providers we considered), and the sender to terminate TLS connections if the recipient's TLS certificate does not validate for the MX's host name (which rarely happens). Whenever unencrypted SMTP is offered as a fallback ($\mathsf{nTLS}^{\mathsf{snd}}(d)$), e.g., with STARTTLS, but typically also with SMTPS, the attacker can compromise the confidentiality of the communication between two mail providers by resolving a provider's MX record to a domain under her control.

$$\frac{\begin{array}{c}d,e\in\mathsf{Provider} \quad d\neq e \quad d\xrightarrow{\text{MX}}d'\\ \mathsf{nTLS}^{\mathsf{snd}}(d) \quad \mathsf{I}^{\mathsf{DNS}}(e)\vee\mathsf{I}^{\mathsf{DNS}}(d',e)\end{array}}{\mathsf{unconf}(d,e)}$$

2. We assume that the domain's resolver is validating and that its validation is trusted in iterative resolution. This is the most common setup.

The predicate $\mathsf{nTLS}^{\mathsf{snd}}(d)$, indicating that d does not enforce strict host validation when sending emails, is initially always false – currently no mail provider does this, the original STARTTLS proposal even considered domain validation a 'local matter' [4]. Still, some popular domains fail strict certificate validation [23]. Consequently, the attacker can modify the IP to which the MX entry (which is a domain) itself is resolved.

$$\frac{\begin{array}{c} d,e\in\mathsf{Provider} \quad d\neq e \quad d\xrightarrow{\text{MX}}d' \quad e\xrightarrow{\text{MX}}e' \\ \mathsf{I}^{\mathsf{DNS}}(e')\vee\mathsf{I}^{\mathsf{DNS}}(d',e') \quad \mathsf{nTLS}^{\mathsf{snd}}(d) \end{array}}{\mathsf{unconf}(d,e)}$$

Routing attacks also apply to TLS but can be countered by proper certificate validation, and also by using DANE or SMTP STS (cf. Section 4).

$$\frac{\begin{array}{c} d,e\in\mathsf{Provider} \quad d\neq e \quad d\xrightarrow{\text{MX}}d' \quad e\xrightarrow{\text{MX}}e' \\ \mathsf{I}^{\mathsf{R}}(d',e') \quad \mathsf{nTLS}^{\mathsf{snd}}(d) \quad \mathsf{nDANE}^{\mathsf{rcv}}(e) \end{array}}{\mathsf{unconf}(d,e)}$$

Even enforcing strict domain validation allows an attacker to intercept emails, as the integrity of the domain name resolution procedure is required to identify the MX responsible for a mail domain. Most client certificates contain the domain name of this MX server, but not the mail domain. Consequently, a connection between two mail domains can still remain unconfidential, even if TLS is enforced, if the adversary is able to refer communication to another server by means of the MX entry. In 2016, the server identity check procedure was updated with RFC 7817: a valid certificate now needs to contain the email domain (domain id, the last part of the email address), as well as the MX domain [22]. This prevents the above attack, hence the attacker action in our model only applies if the mail provider performs no mail domain verification according to RFC 7817 ($\mathsf{nRFC7819}(d)$). None of the mail providers we encountered enforced this strict validation.

$$\frac{\begin{array}{c} d,e\in\mathsf{Provider} \quad d\neq e \quad d\xrightarrow{\text{MX}}d' \\ \mathsf{I}^{\mathsf{DNS}}(e)\vee\mathsf{I}^{\mathsf{DNS}}(d',e) \quad \mathsf{nRFC7819}(d) \end{array}}{\mathsf{unconf}(d,e)}$$

4. Defender Model

We describe the defender model in terms of defender actions that can be composed to mitigation strategies to minimize the attacker's objective value $\mathcal{R}$. In contrast to attacker actions, each defender action f is associated with a cost $c(f) \in \mathbb{R}_0^+$ (the cost from the defender's point of view). We estimate this cost based on publicly available data (our analysis algorithm performs well, cf. Section 6; state actors with access to detailed information will be able to refine our cost estimates and obtain more accurate results). The actual cost to a government are of political nature and depend on whether mitigations are recommended by government agencies, required by regulatory standards or enforced by direct government intervention. We instead focus on the direct economical cost they impose on the implementing party, i.e., the service provider, in order to provide a result with a clear interpretation.

4.1. Routing mitigations

Routing-level attacks affect both communication and name resolution. We thus consider securing packets routed between two ASes via IPSEC. We add a rule permitting the defender to set the predicate $\mathsf{nVPN}(a,b)$ to false, for any two ASes located on its soil, which prevents routing attacks as discussed in Section 3.2.2.

$$\frac{a,b\in\text{AS} \quad a,b\xrightarrow{\text{LOC}} cn_D}{\neg\mathsf{nVPN}(a,b)} \; c=\$56\,000$$

Cost estimation. Public data on AS-level throughput suggest that the vast majority of ASes are connected with a link speed of 10Gbit/s, hence we estimate the cost of providing this throughput at peak times [24]. The computational requirements are well understood, two dedicated routers (one for each side) for $\$24\,000$ each are sufficient [25]. Adding about 80 consulting hours for configuration and testing [26], and annual cost for support and maintenance [26], we arrive at an estimate of $80\cdot\$100+\$48\,000=\$56\,000$ for the cost of deploying a VPN connection. We neglect the energy cost, as an existing router would likely be replaced, and the additional computational cost is almost exclusively symmetric cryptography, which is very efficient on these devices.

4.2. DNS-level mitigations

We consider two countermeasures against attacks to domain resolution: DNSSEC and DANE. DNSSEC was designed to provide, among other properties, origin authentication and data integrity for DNS data. Let $d\xrightarrow{\text{NS}}e$ be given for any name server e that is authoritative for d. DNSSEC mitigates DNS-level attacks that rely on intercepting packets between a mail provider and its resolver or between the resolver and an authoritative name server. It does not provide protection against authoritative name servers under adversarial control. As deployment of DNSSEC on the root servers and the defender countries' TLDs is completed, this mitigation is available as of now. We consider the employment cost per company, hence all servers for which a provider's name server is authoritative are mitigated in one step. As a side-effect, mail providers which are part of the same company share this cost if the same name server is authoritative for them.

$$\frac{\begin{array}{c} f\in\mathsf{Provider} \\ \{d_1,\ldots,d_l\}=\{d\in\text{Dom}\mid d\xrightarrow{\text{DNS}}e\wedge f\xrightarrow{\text{NS}}e\} \end{array}}{\neg\mathsf{nDNSSEC}(d_1)\wedge\cdots\wedge\neg\mathsf{nDNSSEC}(d_l)} \; c=\$366\,342$$

DNSSEC by itself is only useful against packet injection attacks on resolution, as any name server consulted during resolution of a domain is part of the trust chain. But the integrity it provides can be used to indicate that the communication partner prefers communication via TLS. This side-steps the deployment issue that strict certificate validation in TLS faces. There are two technologies which can be used to this end, DANE (DNS-Based Authentication of Named Entities [5]) and SMTP STS (Strict Transport

Security [6]). Both have different aims and approaches: SMTP STS allows mail transfer agents (MTA) to publish their intent to use TLS, e.g., via text records in DNS, while DANE first and foremost transmits information about which certificates should be accepted, e.g., by publishing the hash of the end-to-end certificate. Thus DANE may be used to indicate that TLS shall be used for mail communication by providing the certificate's hash. While DANE can be used to take over some tasks of the public-key infrastructure (PKI), in our model the PKI is fully trusted. Hence with respect to our threat model, both technologies are providing essentially the same functionality.

$$\frac{f \in \mathsf{Provider} \quad f \xrightarrow{\text{NS}} e \quad d \xrightarrow{\text{DNS}} e \quad \neg\mathsf{nDNSSEC}(d)}{\neg\mathsf{nDANE}^{\mathsf{rcv}}(f)} \; c = \$4\,000$$

Cost estimation. We assume all mail providers have a DNSSEC-aware resolver, as software is readily available to this end, and consider only the cost at the receiver's side, where the DNS infrastructure needs to be upgraded. Based on an extensive survey [27] among 19 companies about their capital expenditures for upgrading their infrastructure to handle DNSSEC, we estimate the cost at €335 590 ($366 342), which email providers as large zone operators have to carry.

Deploying DANE or SMTP STS, we consider reconfiguration of services a task that needs to be carried by external consultants. We estimate the time for planning and implementing at 40 working hours, and thus fix the cost for reconfiguration to be $c = 40 * \$100 = \$4\,000$.

4.3. Mitigation by adoption of secure standards

As discussed in Section 3, the connection between two mail providers can be secured by using TLS, however, the efficacy hinges on the way domain validation is handled. All large email providers employ opportunistic SMTPS or STARTTLS, hence for an active adversary, using TLS without any domain validation is immediately vulnerable to packet injection and DNS spoofing attacks. We thus consider the adoption of SMTPS with *a*) validation of the server's domain, and *b*) validation of both the server's domain and the domain part of the email address, i.e., according to RFC 7819.

$$\frac{d \in \mathsf{Provider}}{\neg\mathsf{nTLS}(d)} \; c = \$4k \qquad \frac{d \in \mathsf{Provider}}{\neg\mathsf{nTLS}(d) \wedge \neg\mathsf{nRFC7819}(d)} \; c = \$4\,080$$

Cost estimation. Similar to the adoption of DANE or SMTP TLS, strict enforcement of TLS should not need more than 40 working hours by external consultants, i.e., $40 * 100 = \$4\,000$. The additional running cost imposed by cryptography are negligible. First, the additional costs for the record protocol are very cheap [28], second, modern mail transfer agents support connection sharing, which is very efficient in our setting where a small number of popular mail providers dominates the market. Third, as about 80% to 90% of communication enjoys (opportunistic) TLS already, a relatively small amount of traffic is affected. The implementation cost of RFC 7817 are the same as for strict certificate validation, but the certificate format needs to be adapted to this recent standard, which amounts to about $80 for the first year. We will thus also discuss a model reflecting the loss of revenue due to users quitting the provider in frustration. In Section 6.2 we will modify this cost to account for users that leave a service because emails fail to arrive due to this validation.

4.4. Mitigation by relocation

If the adversary controls infrastructure used by the defender, e.g., the mail server itself, the only remedy is to set up trustworthy infrastructure or relocate it to put it under a different jurisdiction.

$$\frac{e \in \text{Dom} \quad \{d_1, \cdots, d_l\} = \{d \mid d \xrightarrow{\text{DNS}} e\}}{d_1 \not\xrightarrow{\text{DNS}} e \wedge \cdots \wedge d_l \not\xrightarrow{\text{DNS}} e} \; c = \$10\,000$$

$$\frac{d \in \mathsf{Provider} \quad \{e_1, \cdots, e_l\} = \{e \mid d \xrightarrow{\text{MX}} e\}}{d \not\xrightarrow{\text{MX}} e_1 \wedge \cdots \wedge d \not\xrightarrow{\text{MX}} e_l} \; c = \$1 \cdot u(d)$$

Relocation is the most invasive measure considered here, but vis-à-vis the 2015 decision about the safe harbour agreement between the EU and the US, it is not completely unrealistic, as the underlying conflict still remains unresolved [29]. We are excluding authoritative name servers for top-level domains (TLDs) from relocation, but name servers below can be relocated, removing all dependencies to the compromised TLD.

Cost estimation. While relocation cost depend a lot on a company's location, we aim for a uniform treatment which only takes into account the number of users, $u(d)$, a provider d has, ignoring the difference in cost of running the servers at the new location, e.g., rent, electricity, insurance, tax, etc.

Planning and executing the relocation amounts to ca. $10,000 per rack [30]. This gives the cost per name server relocation. For mail servers, the relocation costs scale with the users, hence we multiply these cost with the number of users, as one rack can serve 10k users [31]. As relocation costs are similar to moving the same infrastructure to a (trusted) cloud service [32], we do not distinguish the two scenarios.

5. Data acquisition

In order to evaluate our mitigation analysis, we select attacker and defender countries we consider relevant. We use market share data to determine the most popular email service providers in each defender country. Then we acquire DNS and routing data to derive the property graph described in Section 2.3 and instantiate the threat model.

Country	Internet users	Email	Probes	Server
China	731 434 547	87%	×	✓
India	462 124 989	68%	×	×
United States	286 942 362	88%	×	✓
Brazil	122 796 320	80%	×	✓
Japan	118 131 030	75%	×	×
Russia	105 311 724	86%	×	✓
Nigeria	86 436 611	—	×	×
Mexico	72 945 992	85%	×	×
Germany	70 675 097	89%	✓	✓

TABLE 3: Countries by number of Internet users [39], [40].

Attacker countries. We have to choose which country is taking the 'attacker' role, but stress that this word is to be interpreted in the information security sense. We chose the following criteria to avoid politicising this decision: As we are interested in countries which are both capable and likely to engage in large-scale email sniffing, we consider the purported spending on intelligence services [33], military spending [34], and the press freedom index [35], the first two as indicators for the intelligence capabilities, and the third for the plausibility of such an endeavour.

We consider the seven countries which are both in the top 10 w.r.t. spending on intelligence and military and, in addition, in the bottom 140 of the press freedom index. We also consider the so-called Five Eyes agreement (Australia, Canada, New Zealand, the United Kingdom and the United States) as this alliance is known to engage in email-sniffing [36], as well as the Fourteen Eyes agreement, consisting of Five eyes and Denmark, France, the Netherlands, Norway, Germany, Belgium, Italy, Spain and Sweden. We thus consider the United States, Japan, China, Russia, Italy, Mexico, South Korea, having the role of the attacker, as well as Five Eyes and Fourteen Eyes.

Defender countries & vantage points. Table 3 shows the nine countries with the most Internet users (in absolute numbers). The Internet as a whole, and domain resolution in particular, looks different when observed from different countries, e.g., due to CDNs (e.g., DNS-based request-routing [37]), Anycast DNS [38] and censorship. Unfortunately, we could not get access to machines in certain countries, therefore we chose to omit them. This resulted in the selection of: China, USA, Brazil, Russia and Germany. At each of those vantage points, we instantiate a new property graph with the data gathered from the respective server. We furthermore obtained market share data to evaluate attack impacts and identify the most popular email service providers in the defender countries, see, e.g., Figure 3.

DNS related data. Our data acquisition starts with DNS queries to collect DNS records relevant to our threat model, namely, A records (mapping domain name to an IP), MX records (domain name hosting the email transfer agents used by the domain) and NS records (authoritative name servers used by that domain). We add nodes and relationships (see Table 1) corresponding to these records to the property graph (see Section 2.3). To identify the local DNS resolver (or the DNS forwarder) used by mail servers for name resolution, we monitor the inbound traffic at an external name server, to identify the IP address of the resolver. To this end, we set up a mail server and added its domain name as an MX record into our DNS zone file. Next, we created email accounts with the mail providers that are part of the analysis and accordingly sent emails to our mail server. To avoid cached results that the local DNS resolver/forwarder might have for our domain, we change the domain name of our mail server for every mail provider by using a cache-busting nonce: emails are sent to `resolver@`x`.ourdomain.com`, where x is different for each email. The name server then registers which resolver requested x`.ourdomain.com`.

Routing information and countries. To be able to model AS-level routing attacks and their mitigation, we need routing data for the ASes where the mail providers and name servers are hosted. We chose to measure actual routes, rather than simulating routing policies (e.g., Gao-Rexford) using public peering information. This improves the accuracy of our routing data, however, Germany is the only country of our selection for which we were able to get sufficiently many probes within these ASes. Our methodology can be easily adapted to the simulation approach.

For collecting routing information, we start with identifying the ASes of the mail servers by querying the RIPEStat database [41]. We collect the information about the AS-level routing between two mail servers using the RIPE Atlas [42] network of probes to acquire traceroutes between the mail servers. We thus identify the actual AS-level routes package between two MTAs would take (within the time frame of our measurements). However, not all ASes host a RIPE probe, hence our analyses do not cover all possible pairs of MTAs. Table 3 indicates where our data was insufficient, and hence packet injection capabilities of the attacker were underestimated.

Finally, in order to establish a relationship between the servers and the countries where they belong, we add geolocation data to our property graph. To that end, we query the MaxMind [43] dataset and accordingly create a node of type Cntry along with an edge of type LOC that links the IP address to the selected country.

6. Results and Evaluation

We now apply our methodology for the threat and defender model described in Sections 3 and 4 to the data acquired in Section 5. First, we consider different cost scenarios within a case study (Five Eyes versus Germany) to discuss deployment hazards and the impact of DNSSEC. Then we discuss the results for all combination of defenders and attackers we considered in our analysis. We visualize our results in several figures with a symlog-scaled x-axis which is linear around 0 and logarithmic o/w, the y-axis is scaled linearly. Additionally, we release our source code along with an interactive visualization of the results at [44]. After pre-processing, most of the generated instances could be solved on an Intel Xeon E5-4650L machine within one

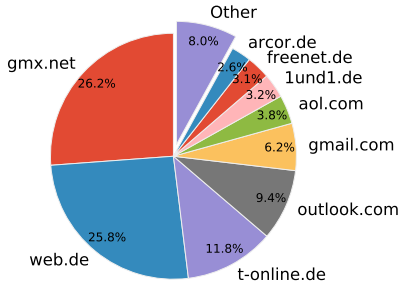

Figure 3: Most used (primary) email providers in Germany [45].

minute. The largest instance required approximately 16 minutes of CPU time. The pre-processing time was dominated by IO operations, and otherwise it was negligible.

6.1. Threats to validity

Provided our formal threat model is correct, we identify the following threats to validity: as mentioned before, we consider only direct monetary cost and treat defender cost uniformly regardless of country and size of company. We rely on email market share studies to estimate the impact of an attack. These consider only user's primary email addresses. The cost for TLS deployment in S3 to S6 are highly speculative. Finally, we were only able to measure routes between ASes for Germany, not for the other countries.

6.2. Case study: Five Eyes versus Germany

We consider the case of the Five Eyes alliance versus Germany to discuss the effects of large-scale email sniffing on a market where foreign companies provide plenty of infrastructure (as opposed to China) but domestic companies still serve the majority of users (unlike Brazil, see Figure 3). Germany has a sizeable market of approx. 62.9 million email users, and it is well covered by RIPE Atlas for collecting routing information (2114 RIPE Atlas probes, 1056 online at the time of writing). We obtained 1 332 594 routes, 176 of which are relevant to this attack scenario. There are three US companies among the top 3 mail providers in Germany; Microsoft, Google and AOL. GMX, web.de and 1&1 are part of the same company and thus share some infrastructure, e.g., they are within the same autonomous system (AS8560). These providers are also the only providers which employ DNSSEC and DANE.

Barring any mitigations, the attacker gets hold of 48 provider to provider connections, representing 45.43% of German user to user communications. Our exposition will follow the cost scenarios detailed in Table 4, which we will motivate as we go along.

S1: Unit cost. In the unit cost scenario, each mitigation has a cost of 1, hence the defender's goal is minimizing the number of measures taken. In this first analysis (see Figure 4), we observe first that the attacker reward can actually be reduced to zero by enforcing TLS connections and RFC 7817 on

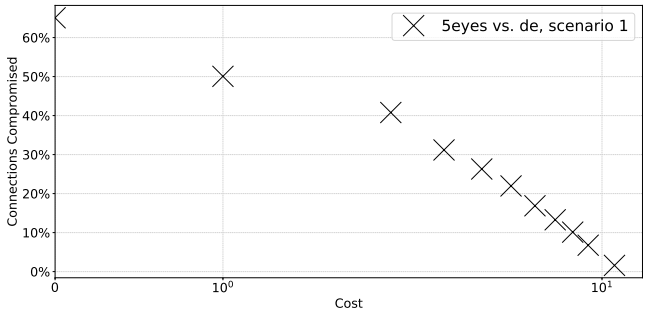

Figure 4: Five Eyes vs. Germany (S1)

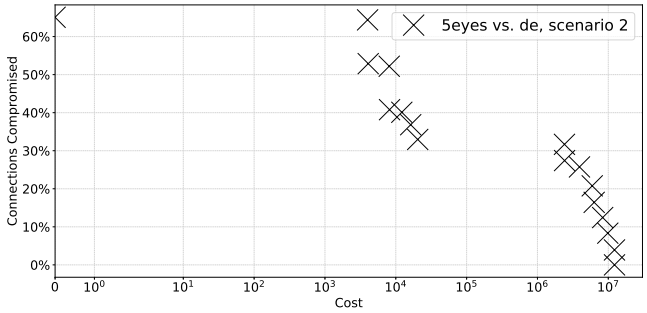

Figure 5: Five Eyes vs. Germany (S2)

all providers and relocating `outlook.com`, `aol.de` and `gmail.com`. As TLS and RFC 7817 validate the identity and the authority of the recipient MX, no relocation for name servers is necessary. Relocation is restricted to foreign mail providers. All other mitigation strategies that result in zero attacker reward are dominated by this mitigation strategy, as TLS and RFC 7817 provide a cure-for-all in two steps, whereas relocation of name servers or VPNs requires mitigation at several points of the network.

S2: actual spending. Now we consider the actual cost spent on implementing these mitigations (see Figure 5). Here again, enforcing TLS and RFC 7817 is the prevalent solution in low-cost and high-security settings. US-based mail providers are proposed to relocate their MXes. RFC 7817 is indeed more useful than the relocation of DNS servers. DNS relocation is rarely considered, as all German providers rely only on infrastructure within Germany, whereas mail servers of US providers need to be relocated anyway.

It thus seems that the deployment of TLS with strict certificate validation according to RFC 7817 is the cheapest way of countering large-scale email sniffing. However, this contradicts the observation that even despite Google's and Facebook's push for encrypted server to server communication, no large mail provider dares to require TLS for SMTP connections.

S3: deployment cost, pessimistic. Opportunity cost is the main obstacle for enforcing TLS, as users leaving a service because they cannot reach their friends or business partners produces a loss in revenue.We consider the average revenue

scenario	routing mit. VPN	DNS mitigations DNSSEC	DANE/SMTP STS	secure standards enforce TLS	RFC 7817	relocation MX	NS
S1: unit cost	1	1	1	1	1	1	1
S2: actual spending	56 000	366 342	4 000	4 000	4 080	1/user	10k
S3: deployment (pessimistic)	56 000	366 342	4 000	4k + c_{pess}/user	(disabled)	1/user	10k
S4: deployment (opt.+pess.)	56 000	366 342	4 000	4k + c_{opt}/user	4 080+c_{pess}/user	1/user	10k
S5 (a,b): enforce TLS (opt./pess.)	56 000	(disabled)	(disabled)	4k + $c_{opt/pess}$/user	(disabled)	1/user	10k
S6 (a,b): RFC 7817 (opt./pess.)	56 000	(disabled)	(disabled)	(disabled)	4 080 + $c_{opt/pess}$/user	1/user	10k
S7: hidden attacker	(like S3, but rule r_{dns-ns} has nDNSSEC(n) in its premise)						

TABLE 4: cost scenarios: all cost in \$. $c_{pess} = 0.73 \cdot 7 \cdot 0.5 = 2.555$, $c_{opt} = 0.0036 \cdot 7 \cdot 0.5 = 0.0126$.

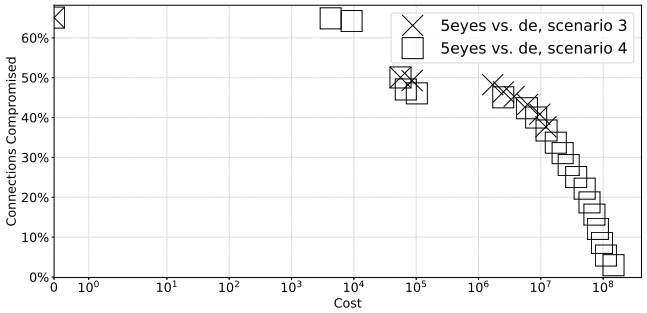

Figure 6: Five Eyes vs. Germany (S3 and S4)

per user (ARPU)[3] , which ranges between \$3.50 and \$20 for most companies [46], [47]. Again opting for a uniform treatment, we chose a conservative estimate of \$7.00, but remark that the ARPU is likely to fluctuate from country to country, and from company to company (Google, e.g., has a considerably higher ARPU than other companies), however, the *potential revenue* from users of different providers is relatively similar [48].

We make the pessimistic estimate that a provider would lose 73% of his users when enforcing TLS (SMTPS or STARTTLS) as opposed to the current opportunistic model. About half of the users are using web-mail [49]; we presume only those to produce revenue, as they receive advertisement and are likely to be active on cloud services. Hence we assume $c_{pess} = 0.73 \cdot \$7.50 \cdot 0.5 \approx \$2.56$ per user are lost if TLS is enforced.

The estimate of 73% corresponds to the following model: Long-running studies show that business users send about 20 and receive about 100 emails per day [50], [51]. As of May 2017, Google reports that 88% of outbound and 86% inbound emails are already passing standard certificate validation [52], Facebook and Yahoo report similar figures. Taking these numbers as a basis for an average user, 73% is the probability that this average user gives up, if she insists that $t_{in} = 90\%$ of inbound emails addressed to her and $t_{out} = 70\%$ of her outbound emails reach their destination.[4] If all users were to behave like this average user, about 0.73% of users would leave a provider.

In this scenario, TLS is altogether avoided as a mitigation (see Figure 6). Throughout the Pareto frontier, only mitigation by relocation appears. The most expensive but effective mitigation is incurring \$15M in cost and requires relocating four name servers of AOL and the mail servers of AOL, Microsoft, Google, 1&1 and Arcor, favoring relocation over the deployment of TLS. This is consistent with the observation that, in the wild, TLS connections are not enforced by mail providers, which is likely because early adopters would be punished by clients leaving for less secure but more functional services. It is remarkable that DANE/SMTP STS *never* appear in the Pareto frontier – as in all other scenarios. Upon closer inspection, this is due the fact that there is only a single route between two MXes (out of 176 relevant routes) that traverses a compromised AS, but where DANE/SMTP STS can be of use because the destination MX is not compromised from the start, e.g., domestic. This route connects the MXes of gmx.net, web.de and 1und1.de (who all belong to United Internet) with t-online. DANE requires setting up a DNSSEC infrastructure, but for t-online, this infrastructure is of little other use (in our model at least), as t-online does not rely on authoritative name servers controlled by the attacker. Hence it is cheaper to set up a VPN for this specific route. Naturally, our routing data is incomplete, but it appears DANE/SMTP STS is of little use when domestic providers are communicating via domestic infrastructure. We disabled RFC 7817, since estimating the cost of its deployment is pure speculation. (We will indulge in speculation in the next section, though.) Consequently, even with an unlimited budget, it is not possible to lower the attacker reward to 0 and 19M transmissions (of $4.95 \cdot 10^{15}$) cannot be considered confidential.

S4: deployment cost, optimistic and pessimistic. We take a more optimistic view on the deployment obstacles of TLS enforcement by considering an average user who is perfectly happy as long as half her incoming emails reach her, and half her outgoing emails reach their recipient. If every provider's user base consisted of only this user, enforcing TLS would cost a provider 3.6‰ of its user base. For RFC 7817, we assume the pessimistic scenario, as it is relatively new (March 2016).

Under these circumstances, enforcing TLS is very useful even in low-cost scenarios (see Figure 6). Up to a total cost of \$25k, the dominant solution is to enforce TLS with the largest subset (by market share) of German mail providers

3. Average margin per user might seem more appropriate, however, it is difficult to obtain figures on the profit of silicon valley companies. Moreover, it is very costly to adopt a decrease in users, hence we assume the cost for infrastructure remains unchanged.

4. We can compute the probability that this average user gives up using two binomial distributions: $p_{gives\,up} := 1 - \Pr[I > t_{in} \cdot n_{in}] \cdot \Pr[O > t_{out} \cdot n_{out}]$, where $n_{in} = 100$, $n_{out} = 20$ and $I \sim B(n_{in}, 0.86)$, $O \sim B(n_{out}, 0.88)$.

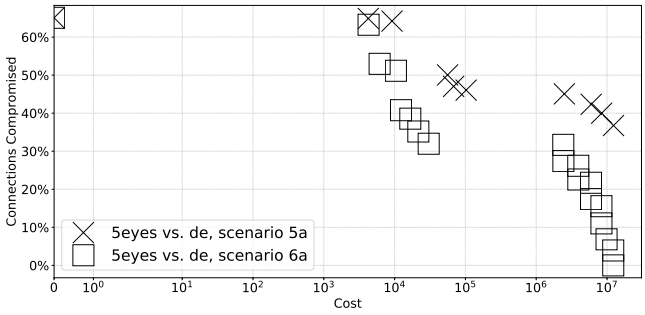

Figure 7: Five Eyes vs. Germany (S5a and S6a)

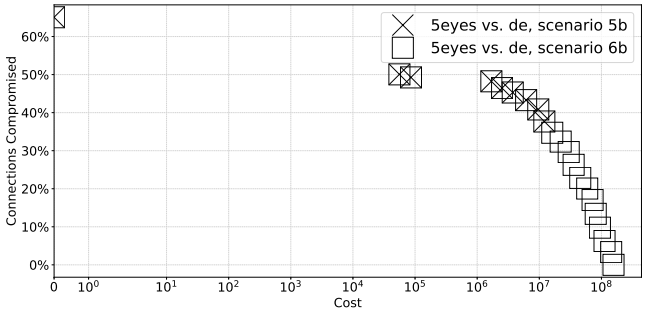

Figure 8: Five Eyes vs. Germany (S5b and S6b)

within the budget. Above that, a mix between enforcing TLS and relocating mail servers and name servers of US mail providers is the most cost-efficient. Enforcing RFC 7817 is only gaining interest starting at a $12M budget. Relocating the mail servers of Microsoft, AOL and Google (but not 1&1 and Arcor, as in S3), the adversarial success can be reduced to 0 if all remaining providers enforce TLS with validation according to RFC 7817, however, at the cost of approx. $156M. This confirms that RFC 7817 validation is an effective countermeasure, even if it comes at high cost.

S5+S6: the security advantage of deploying RFC 7817. The results in scenario S3 and S4 suggest the following question: If one would go through the pain of enforcing TLS for SMTP traffic, would it not make sense to deploy RFC 7817 right away? We compare those two scenarios by disabling one or the other and comparing the results. RFC 7817 is strictly stronger but requires some additional effort at the recipients' side. It is hard to estimate this cost, but our comparison gives an indication of the cost saved by other measures becoming redundant due to RFC 7817's stronger resistance against attacks on the DNS level.

In the optimistic scenario (see Figure 7), our data suggest that not only RFC 7817 is the better mitigation from the start, but starting from a budget of approx. $4 000, the attacker reward is consistently 20 percentage points higher if TLS is enforced (w/o RFC 7817) for the same budget. Again it is not possible to lower the attacker reward below 19M connections, whereas validation according to RFC 7817 can lower it to zero.

In the pessimistic scenario (see Figure 8), however, there is almost no difference between both Pareto frontiers, because in both scenarios, relocation of NS servers and IPsec

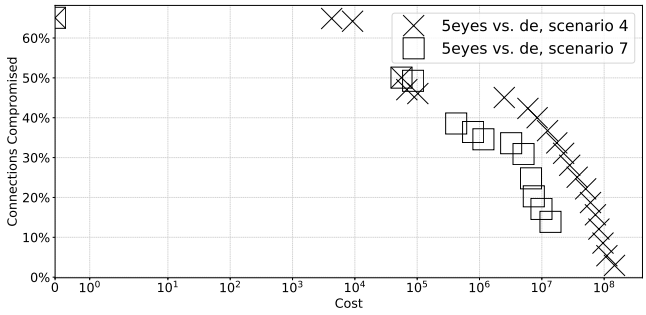

Figure 9: Five Eyes vs. Germany (S4 and S7)

is preferred to enforcing TLS, be it with or without RFC 7817 certificate validation: TLS is never enforced in the Pareto frontier to S5b. TLS+RFC 7817 appears in S6b at the cost of ca. $14M, however, it is only slightly more effective than the corresponding solution by relocation in S5b.

S7: hidden attacker, crouching DNSSEC. DNSSEC can also be used as a forensic tool, as signatures permit identifying misbehaving parties and provide a time stamp to indicate responsible actors [53]. As our motivation is an attacker who attempts large-scale surveillance without getting caught too often, it is interesting to look at the scenario where the integrity of a domain is protected by DNSSEC even if a name server used during resolution (and thus part of the trust chain) is controlled by the adversary, because she is trying to avoid exposure. This amounts to simply adding $\mathsf{nDNSSEC}(n)$ to the premise of rule r_{dns-ns} (see p. 5).

Comparing the Pareto frontier in S4 and S7 (see Figure 9), we observe no difference from the start, but a significant increase of confidential connections in the mid-cost region. The attacker reward is lowered to zero one order of magnitude earlier. With increasing budget, first the IPsec connection for the aforementioned route (see S3) is used, then domestic providers deploy DNSSEC, except for t-online (which has little dependency on compromised name servers) and freenet.de (where it is cheaper to relocate the few name servers they have). As DNSSEC now also mitigates DNS spoofing via authoritative name servers, it becomes a cheaper alternative to strict TLS enforcement at all budgets. This means that DNSSEC is a very cost-efficient and effective counter-measure against large-scale email sniffing, provided that the attacker can be held responsible for misbehavior or has other reasons to avoid exposure that cannot be denied easily. Again, DNSSEC is not deployed, as it is cheaper for t-online to secure the single malicious route via IPsec than to use DANE/SMTP STS and pay the additional cost of deploying DNSSEC.

6.3. Results for other countries

We now turn our attention to the other defender countries. For space reasons, we limit the discussion to the fourth cost scenario (S4), which allows to compare the effect of all defender actions. Figures 10 to 14 give an combined view of the respective defender countries against the different

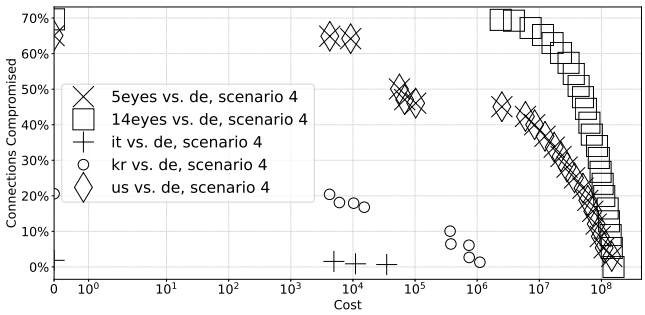

Figure 10: Results for defender country Germany (S4)

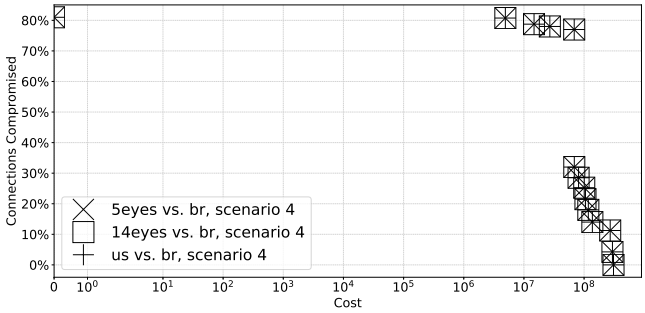

Figure 11: Results for defender country Brazil (S4)

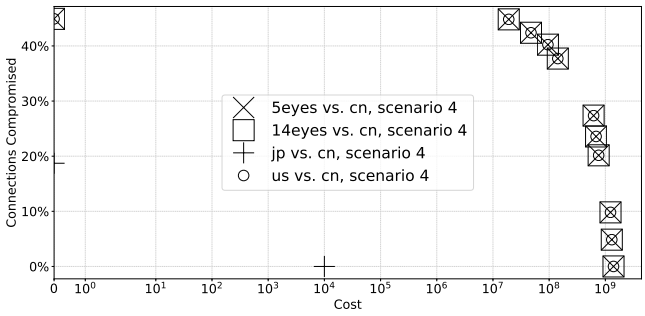

Figure 12: Results for defender country China (S4)

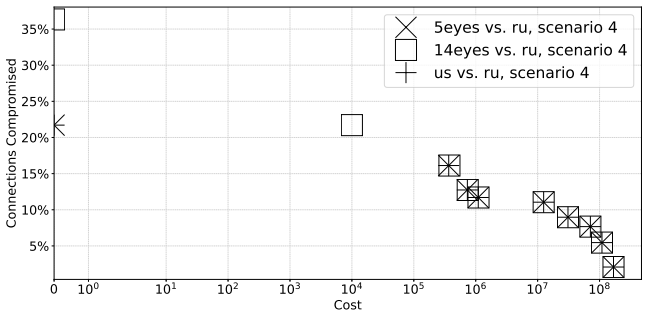

Figure 13: Results for defender country Russia (S4)

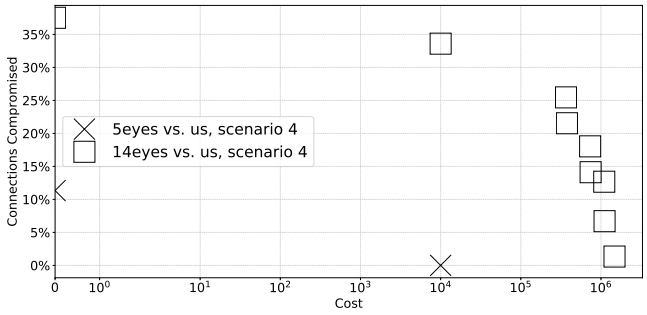

Figure 14: Results for defender country USA (S4)

attacker countries. Attacker countries which cannot compromise any email communication of the considered defender country have been left out, and coalitions are excluding the defending country.

Germany. Among the Five Eyes countries, only the US can affect email confidentiality of German users. In low-cost scenarios below $56k, enforcing TLS provides a remedy, above this budget, and up to $151M, securing the connection between two ASes via IPsec is an effective measure. To completely defend, it is necessary to relocate the mail servers of the American providers, and to enforce TLS with RFC 7819 for all providers to protect from TLD servers hosted in the USA. Extending the attacker to all Fourteen Eyes countries leads to additional attacks on domain resolution, significantly increasing the necessary mitigation budget. Relocating all compromised mail servers, and enforcing TLS with RFC 7819 certificate validation, however, is still sufficient to protect Germany's email confidentiality. Finally, South Korea and Italy are able to compromise email confidentiality through malicious routes, but only to marginal success. For South Korea, only DNSSEC and TLS enforcement appear in the Pareto frontier, for Italy only relocation and TLS enforcement.

Brazil. According to our statistics, US mail providers entirely dominate the email market of Brazil [54]. Consequently, Brazil has a significant dependency on US infrastructure. Relocating mail servers of the less used providers (yahoo and gmail) along with enforcing TLS with RFC 7819 is the cheapest mitigation strategy. Starting at 68M, relocating the email servers of hotmail, being by far the most prominent provider in Brazil, leads to a tremendous decrease in compromised connections.

China. Only the US and Japan are relevant attacking countries in this scenario. To mitigate attacks from Japan, it is enough to relocate a single name server. On the other hand, to mitigate attacks from the US, the best solution is to relocate the mail servers of hotmail and to enforce TLS and RFC 7819 on the domestic providers. Note that DNSSEC seems irrelevant because we lack information about routing, and mostly Chinese TLDs are used.

Russia. Apart from gmail.com, Russian users rely on domestic email providers. This is the reason why there is

no country or alliance that can compromise more than ca. 36%. Relevant attackers are the US and Fourteen Eyes. The only difference between them is that against Fourteen Eyes a single compromised name server needs to be relocated. The immediate and the cheapest mitigation is to activate DNSSEC on the three domestic providers because they use US controlled TLDs. Further, relocating the email servers of gmail and activating DNSSEC on only two domestic providers is a close second. To reach confidentiality for all communication, it is necessary to enable DNSSEC and enforce TLS with RFC 7819 on all domestic providers and to relocate gmail's email servers.

USA. To defend from other countries which are part of Five Eyes, it is necessary for the US to relocate a single name server which resides in the UK. For defending from the Fourteen Eyes countries, more action is needed. Strictly speaking, all five widely used domestic email providers need to enable DNSSEC because some routes are traversing foreign countries.

7. Related work

Infrastructure analysis on the Internet. Many works have been studying global and targeted attacks on the Internet infrastructure and the reasons behind these attacks [55], [56], [20] but the systematic evaluation of mitigations has largely been neglected. Frey et al. [57] studied the European BGP topology w.r.t. to disruption scenarios, but considered only three possible outcomes from mitigation. Simeonovski et. al. [20] presented a model of the Internet infrastructure using property graphs to assess attacker impact, but not possible mitigations. We extend their property graph for modeling the Internet infrastructure and adopt a completely new and more expressive attacker model (adding, e.g., name resolution and routing).

Automated mitigation analysis. As previously mentioned, our work builds on the recent proposal by Speicher et al. [8], introducing automated mitigation analysis in the context of simulated pentesting, i.e., automated security testing for corporate networks. Our planning framework, as well as the associated mitigation analysis algorithms, are extensions of this work, so some words are in order regarding related models. Simulated pentesting is rooted in the consideration of attack graphs, first introduced by Philipps and Swiler [17]. An attack graph breaks down the space of possible attacks into atomic components, often referred to as attack actions. Much as in AI planning, the attack graph is intended as an analysis of threats that arise through the possible combinations of these actions. Different variants of attack graphs provide analyses at different levels of complexity, ranging from simple threat overviews (e.g., [58], [59]) to a full state-space verification (e.g., [60], [61]). A prominent middle ground between the two is the *monotonic* formulation – positive preconditions and postconditions only – that we employ here as well, where attackers keep gaining new assets, but never lose any assets during the course of the attack [59], [62], [63], [64], [65], [18], [66]. A close relative of attack graphs are *attack trees* (e.g., [58], [67]) a form of 'Graphical Security Models' [68]: Directed acyclic AND/OR graphs organizing known possible attacks into a top-down refinement hierarchy. The human user writes that hierarchy, and the computer analyzes how attack costs and probabilities propagate through the hierarchy.

Mitigation analysis models not only the attacker, but also the defender. It thus relates to game-theoretic security models, specifically to Stackelberg competitions, where the game consists of a single exchange of move and countermove. In our setting, each 'move' here consists of an entire (defender- respectively attacker-) action strategy.

The most prominent application of game-theoretic security models thus far concerns physical infrastructures and defenses (e.g., [69]), quite different from the network security setting. A line of research considers attack-defense trees (e.g., [70], [68]), extended Graphical Security Models including defending nodes. Some research considers pentesting from a very abstract theoretical perspective [71].

The work most closely related to ours is that by Durkota et al. [72], [73], [74], [75]. Like our mitigation analysis, Durkota et al.'s work line considers a Stackelberg formulation of security testing. Like Speicher et al.'s work we build on, Durkota et al.'s work is placed in the network security context. Apart from this different application, major differences lie in the attacker and defender models considered as part of the game. On the one hand, on the defender's side, Durkota et al.'s model is limited to the placement of honeypots, modeled as additional fake machines added to the network within *pools* of equivalent machines indistinguishable to the attacker. This is in contrast to our framework which allows general AI planning defender actions, in order to be able to model complex infrastructure modifications. On the other hand, on the attacker's side, Durkota et al.'s model is more general than ours. It considers probabilistic attacker actions, with an execution semantics where attacker actions refer to machine pools, and a concrete machine is chosen randomly. The latter is needed to give meaning to the honeypot placement (if a honeypot happens to be chosen, the attack is stopped immediately). Such complexity is not required, however, in our framework, where deterministic attacker actions suffice for modeling purposes. Algorithmically, the consequence is that Durkota et al.'s work focuses on tackling the complexity of attack planning, while Speicher et al.'s algorithms, that we adopt here, focus on tackling the complexity of search through the space of defender-action strategies.

8. Conclusion

We showed that a holistic analysis of deployment benefits of cryptographic protocols, secure configurations and political measures is possible with a high degree of automation. Based on a very simple cost assessment in the case of email communication, we see that the enforcement of TLS would have a great effect in most countries. If strict TLS validation can be achieved, RFC 7817 should be implemented with it:

despite its simplicity, it reduces mitigation cost by more than 20%. However, it is plausible that the hidden cost of deployment, a loss of functionality, make this approach impractical. Techniques like DANE and SMTP STS do not suffer this problem, however, in the case of Five eyes vs Germany, they are never of use. This might be different for Russia vs USA and US vs Five eyes, where DNSSEC is employed a lot, but the lack of routing data prohibits a conclusion in these cases. DNSSEC itself is particularly useful in the scenario where the attacker does not tamper with the trust chain. There it substitutes enforcement of TLS completely. Such *sneaky* adversaries should receive further analysis in the future. Even if the adversary is not sneaky, DNSSEC is useful when the domestic infrastructure is relatively self-sufficient, e.g., for Russia and the US in scenario S4. While the US, Russia and China are relatively self-sufficient, the privacy of email users in Brazil is highly vulnerable due to foreign dependencies, requiring costly relocation.

An interesting avenue for future work is to capture probabilistic threat model, e.g., off-path attacks on DNS that succeed only with a certain chance, vulnerability assessment of old operating systems, or the chance of a user becoming victim to fishing attacks. Probabilistic planning provides a fitting framework for this endeavour and could help, e.g., to prepare emergency responses to large-scale incidents like the recent spreading of the WannaCry worm. Furthermore, the rules defining our threat model are ad-hoc. Similar to how abstractions of cryptographic primitives in the Dolev-Yao model are justified via computational soundness, it is worth exploring how holistic threat models for risk assessment, which have to be somewhat sound and complete, can be derived from protocol specifications. Most importantly: precise cost estimates are hard to come by, in particular the cost induced by loss of functionality and interoperability. We hope that the methodology presented here encourages the in-depth analysis and validation of deployment cost, both for existing and for future security mechanisms.

References

[1] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, "DNS Security Introduction and Requirements," RFC 4033, IETF, 2005.

[2] R. Arends and R. Austein and M. Larson and D. Massey and S. Rose, "Resource Records for the DNS Security Extensions," RFC 4034, IETF, 2005.

[3] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, "Protocol Modifications for the DNS Security Extensions," RFC 4035, IETF, 2005.

[4] P. Hoffman, "SMTP Service Extension for Secure SMTP over Transport Layer Security," RFC 3207, IETF, 2002.

[5] V. Dukhovni and W. Hardaker, "SMTP Security via Opportunistic DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS)," RFC 7672, IETF, 2015.

[6] D. Margolis, M. Risher, N. Lidzborski, W. Chuang, B. Long, B. Ramakrishnan, A. Brotman, J. Jones, F. Martin, K. Umbach, and M. Laber, "SMTP Strict Transport Security," Internet Engineering Task Force, Internet-Draft, Mar. 2016, work in Progress.

[7] E. Rescorla, "HTTP Over TLS," RFC 2818 (Informational), IETF, 2000.

[8] P. Speicher, M. Steinmetz, M. Backes, J. Hoffmann, and R. Künnemann, "Stackelberg planning: Towards effective leader-follower state space search," in *AAAI'18*, 2018, forthcoming.

[9] I. The Radicati Group, "Email statistics report, 2017-2019," 2017.

[10] M. Ghallab, D. Nau, and P. Traverso, *Automated Planning: Theory and Practice*. Morgan Kaufmann, 2004.

[11] W. Ruml, M. B. Do, R. Zhou, and M. P. J. Fromherz, "On-line planning and scheduling: An application to controlling modular printers," vol. 40, 2011.

[12] A. Koller and J. Hoffmann, "Waking up a sleeping rabbit: On natural-language sentence generation with FF," in *International Conference on Automated Planning and Scheduling*, 2010.

[13] M. Helmert and H. Lasinger, "The scanalyzer domain: Greenhouse logistics as a planning problem," in *International Conference on Automated Planning and Scheduling*, 2010.

[14] M. Boddy, J. Gohde, T. Haigh, and S. Harp, "Course of action generation for cyber security using classical planning," in *International Conference on Automated Planning and Scheduling*, 2005.

[15] J. Lucangeli, C. Sarraute, and G. Richarte, "Attack planning in the real world," in *Workshop on Intelligent Security*, 2010.

[16] J. Hoffmann, "Simulated penetration testing: From "Dijkstra" to "Turing Test++"," in *International Conference on Automated Planning*, 2015.

[17] C. Phillips and L. P. Swiler, "A graph-based system for network-vulnerability analysis," in *New Security Paradigms Workshop*, 1998.

[18] N. Ghosh and S. K. Ghosh, "An intelligent technique for generating minimal attack graph," in *Workshop on Intelligent Security*, 2009.

[19] T. Bylander, "The computational complexity of propositional STRIPS planning," 1994.

[20] M. Simeonovski, G. Pellegrino, C. Rossow, and M. Backes, "Who controls the internet?: Analyzing global threats using property graph traversals," in *International Conference on World Wide Web*, 2017.

[21] G. Greenwald, "Xkeyscore: Nsa tool collects 'nearly everything a user does on the internet'," *The Guardian*, Jul. 2013.

[22] A. Melnikov, "Updated Transport Layer Security (TLS) Server Identity Check Procedure for Email-Related Protocols," RFC 7817, IETF, 2016.

[23] M. Adkins, "The current state of smtp starttls deployment," 2014. [Online]. Available: https://de-de.facebook.com/notes/protect-the-graph/the-current-state-of-smtp-starttls-deployment/1453015901605223/

[24] PeeringDB, "De-cix frankfurt," 2017. [Online]. Available: https://www.peeringdb.com/ix/31

[25] D. Raumer, S. Gallenmüller, P. Emmerich, L. Märdian, and G. Carle, "Efficient serving of vpn endpoints on cots server hardware," in *Cloud Networking*, 2016.

[26] M. Chapple, "How expensive are ipsec vpn setup costs?" 2017. [Online]. Available: http://searchsecurity.techtarget.com/answer/How-expensive-are-IPsec-VPN-setup-costs

[27] E. Network and I. S. Agency, "The cost of dnssec deployment," 2010.

[28] I. Grigorik, *High Performance Browser Networking: What every web developer should know about networking and web performance*, 1st ed. O'Reilly Media, Sep. 2013.

[29] European Data Protection Supervisor, "Privacy shield: more robust and sustainable solution needed," 2016.

[30] Info Tech Research Group, "Data Center & Facilities Optimization," https://www.infotech.com/research/ss/consolidate-data-centers, 2017.

[31] Sophos Ltd., "Dedicated sophos email appliances," 2017. [Online]. Available: https://www.sophos.com/en-us/products/secure-email-gateway/tech-specs.aspx

[32] J. Shirman, "Cloud, server migration and price elasticity," 2014. [Online]. Available: http://www.rivermeadow.com/blog/cloud-server-migration-and-price-elasticity

[33] C. Hippner, "A study into the size of the world's intelligence industry," Ph.D. dissertation, Mercyhurst College, Pennsylvania, 2009.

[34] S. I. P. R. Institute, "Trends in world military expenditure," 2016, retrieved 24 April 2017.

[35] R. W. Borders, "World press freedom index 2016," 2016, retrieved 03 May 2017.

[36] Norddeutscher Rundfunk, "Snowden-interview: Transcript," Jan. 2014.

[37] M. Wang, P. P. Jayaraman, R. Ranjan, K. Mitra, M. Zhang, E. Li, S. U. Khan, M. Pathan, and D. Georgakopoulos, "An overview of cloud based content delivery networks: Research dimensions and state-of-the-art," *Trans. Large-Scale Data- and Knowledge-Centered Systems*, vol. 20, 2015.

[38] J. Abley and K. Lindqvist, "Operation of Anycast Services," RFC 4786, IETF, 2006.

[39] Internet Live Stats, "Internet users by country," 2016. [Online]. Available: http://www.internetlivestats.com/internet-users-by-country/

[40] G. Duncan, 2012. [Online]. Available: https://www.digitaltrends.com/mobile/believe-it-or-not-email-is-still-the-killer-app/

[41] RIPE Stat, "Information about specific IP addresses and prefixes," https://stat.ripe.net/.

[42] RIPE Atlas, "Internet data collection system," https://atlas.ripe.net/, 2017.

[43] MaxMind, "IP Geolocation and Online Fraud Prevention," http://dev.maxmind.com/, 2017.

[44] P. Speicher, M. Steinmetz, R. Künnemann, M. Simeonovski, G. Pellegrino, J. Hoffmann, and M. Backes, "Source code and interactive visualization of the results." 2017. [Online]. Available: http://mitigations.whocontrolstheinternet.com

[45] C. C. GmBH, "Studie zur mobilen e-mail-nutzung in deutschland," 2016, survey took place in December 2015.

[46] T. Louis, "How much is a user worth?" 2013. [Online]. Available: https://www.forbes.com/sites/tristanlouis/2013/08/31/how-much-is-a-user-worth/#496672781c51

[47] A. Schiff, "Facebook made almost 20 dollars in average revenue per user in q4 a big jump," 2016. [Online]. Available: https://adexchanger.com/platforms/facebook-made-almost-20-average-revenue-per-user-q4-big-jump/

[48] C. Lancellotti-Young, "The new gmail: +1 for revenue, -1 for opens," 2013. [Online]. Available: http://www.sailthru.com/marketing-blog/the-new-gmail-1-for-revenue-1-for-opens/

[49] K. Lewkowicz, "2017 state of email report," 2017.

[50] I. The Radicati Group, "Email statistics report, 2015-2019," 2015.

[51] G. für Konsumforschung, "Die große office-studie 2014," 2014.

[52] G. Inc., "Email encryption in transit," 2017. [Online]. Available: https://www.google.com/transparencyreport/saferemail

[53] H. Shulman and M. Waidner, "Towards forensic analysis of attacks with dnssec," in *2014 IEEE Security and Privacy Workshops*, 2014.

[54] P. Vahabi, 2013. [Online]. Available: https://www.quora.com/Who-are-the-most-popular-email-client-providers-in-India-and-Brazil

[55] S. Goldberg, "Why is it taking so long to secure internet routing?" *Commun. ACM*, vol. 57, 2014.

[56] K. R. B. Butler, T. R. Farley, P. D. McDaniel, and J. Rexford, "A survey of BGP security issues and solutions," *Proceedings of the IEEE*, vol. 98, 2010.

[57] S. Frey, Y. Elkhatib, A. Rashid, K. Follis, J. Vidler, N. J. P. Race, and C. Edwards, "It bends but would it break? topological analysis of BGP infrastructures in europe," in *IEEE European Symposium on Security and Privacy*, 2016.

[58] B. Schneier, "Attack trees," *Dr. Dobbs Journal*, 1999.

[59] S. J. Templeton and K. E. Levitt, "A requires/provides model for computer attacks," in *New Security Paradigms Workshop*, 2000.

[60] R. W. Ritchey and P. Ammann, "Using model checking to analyze network vulnerabilities," in *IEEE Symposium on Security and Privacy*, 2000.

[61] O. Sheyner, J. W. Haines, S. Jha, R. Lippmann, and J. M. Wing, "Automated generation and analysis of attack graphs," in *IEEE Symposium on Security and Privacy*, 2002.

[62] P. Ammann, D. Wijesekera, and S. Kaushik, "Scalable, graph-based network vulnerability analysis," in *ACM Conference on Computer and Communications Security*, 2002.

[63] S. Jajodia, S. Noel, and B. O'Berry, "Topological analysis of network attack vulnerability," in *Managing Cyber Threats: Issues, Approaches and Challenges*, 2005.

[64] X. Ou, W. F. Boyer, and M. A. McQueen, "A scalable approach to attack graph generation," in *ACM Conference on Computer and Communications Security*, 2006.

[65] S. Noel, M. Elder, S. Jajodia, P. Kalapa, S. O'Hare, and K. Prole, "Advances in topological vulnerability analysis," in *Cybersecurity Applications & Technology Conference for Homeland Security*, 2009.

[66] H. Kil, S. Oh, E. Elmacioglu, W. Nam, and D. Lee, "Graph theoretic topological analysis of web service networks," *World Wide Web*, vol. 12, 2009.

[67] S. Mauw and M. Oostdijk, "Foundations of attack trees," in *Information Security and Cryptology*, 2005.

[68] B. Kordy, P. Kordy, S. Mauw, and P. Schweitzer, "ADTool: security analysis with attack-defense trees," in *Quantitative Evaluation of Systems*, 2013.

[69] M. Tambe, *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press, 2011.

[70] B. Kordy, S. Mauw, S. Radomirovic, and P. Schweitzer, "Foundations of attack-defense trees," in *Formal Aspects in Security and Trust*, 2010.

[71] R. Böhme and M. Félegyházi, "Optimal information security investment with penetration testing," in *The 1st International Conference on Decision and Game Theory for Security*, 2010.

[72] K. Durkota, V. Lisy, C. Kiekintveld, and B. Bosansky, "Optimal network security hardening using attack graph games," in *International Joint Conference on Artificial Intelligence*. AAAI Press/IJCAI, 2015.

[73] Durkota, Karel and Lisy, Viliam and Kiekintveld, Christofer and Bosansky, Branislav, "Game-theoretic algorithms for optimal network security hardening using attack graphs," in *International Conference on Autonomous Agents and Multiagent Systems*, 2015.

[74] K. Durkota, V. Lisý, B. Bosanský, and C. Kiekintveld, "Approximate solutions for attack graph games with imperfect information," in *International Conference on Decision and Game Theory for Security*, 2015.

[75] K. Durkota, V. Lisý, C. Kiekintveld, B. Bosanský, and M. Pechoucek, "Case studies of network defense with attack graph games," *IEEE Intelligent Systems*, vol. 31, 2016.