Retroactive Identification of Targeted DNS Infrastructure Hijacking

Gautam Akiwate
UC San Diego
gakiwate@cs.ucsd.edu

Raffaele Sommese
University of Twente
r.sommese@utwente.nl

Mattijs Jonker
University of Twente
m.jonker@utwente.nl

Zakir Durumeric
Censys/Stanford University
zakir@cs.stanford.edu

KC Claffy
CAIDA/UC San Diego
kc@caida.org

Geoffrey M. Voelker
UC San Diego
voelker@cs.ucsd.edu

Stefan Savage
UC San Diego
savage@cs.ucsd.edu

ABSTRACT

In 2019, the US Department of Homeland Security issued an emergency warning about *DNS infrastructure tampering*. This alert, in response to a series of attacks against foreign government websites, highlighted how a sophisticated attacker could leverage access to key DNS infrastructure to then hijack traffic and harvest valid login credentials for target organizations. However, even armed with this knowledge, identifying the existence of such incidents has been almost entirely via post hoc forensic reports (*i.e.*, after a breach was found via some other method). Indeed, such attacks are particularly challenging to detect because they can be very short lived, bypass the protections of TLS and DNSSEC, and are imperceptible to users. Identifying them retroactively is even more complicated by the lack of fine-grained Internet-scale forensic data. This paper is a first attempt to make progress at this latter goal. Combining a range of longitudinal data from Internet-wide scans, passive DNS records, and Certificate Transparency logs, we have constructed a methodology for identifying potential victims of sophisticated DNS infrastructure hijacking and have used it to identify a range of victims (primarily government agencies), both those named in prior reporting, and others previously unknown.

CCS CONCEPTS

• **Networks** → **Naming and addressing**; • **Security and privacy** → **Security protocols**; *Web protocol security*.

ACM Reference Format:
Gautam Akiwate, Raffaele Sommese, Mattijs Jonker, Zakir Durumeric, KC Claffy, Geoffrey M. Voelker, and Stefan Savage. 2022. Retroactive Identification of Targeted DNS Infrastructure Hijacking. In *ACM Internet Measurement Conference (IMC '22), October 25–27, 2022, Nice, France*. ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3517745.3561425

1 INTRODUCTION

Sophisticated attackers use a variety of methods to gain access into organizations. These methods can include spear phishing (*e.g.*, the Democratic National Committee [62]), abusing software vulnerabilities (*e.g.*, Equifax [8]), or exploiting weak passwords (*e.g.*, SolarWinds [42]). However, a less widely-appreciated vector involves the careful manipulation of DNS infrastructure *in order to acquire valid login credentials or session tokens* to a targeted organization.

This paper focuses on such a class of attack in which an adversary has obtained the capability to manipulate a target domain's DNS configuration. This capability is typically obtained by compromising either the domain holder's account with its registrar or compromising the registrar itself, although we are also aware of versions of the attack that involve compromising accounts at DNS nameserver hosting providers. Using this capability, an attacker can temporarily divert a domain's traffic in order to pass the domain validation check of Certificate Authorities (CAs) such as Let's Encrypt or Comodo to obtain a TLS certificate. Having obtained a CA-signed TLS certificate, attackers then — at a time of their choosing — can arrange to divert traffic for specific subdomains that host TLS-protected services requiring cleartext user credentials (*e.g.*, SMTP, VPN, IMAP, etc.). The attacker can then extract any such credentials as users interface with these services, and can repurpose them for further access inside the organization.

Versions of such attacks, in use by state-affiliated actors, date back to 2013 [10, 58] but they became much more widely known in early 2019 when Cisco Talos [43] and FireEye's Mandiant [33] documented particular attacks and victims in the Middle East. This led the US DHS to issue an emergency directive about the threat and mandate a range of mitigations on government systems [20]. However, identifying the victims of such attacks was left to each organization since it relied upon their individual diligence and site-specific knowledge (*e.g.*, in auditing DNS records and validating issued TLS certificates for their domains). The challenge of third-party auditing, along with the short time scales over which such attacks can operate, perhaps explains why there have been limited investigations of this threat and its victims.

This paper sets out to explore this question empirically and retroactively, identifying domains that may have been hijacked in this manner and focusing on those cases that are likely to represent real victims. Using four years of longitudinal data across multiple data sources, including certificate transparency logs, passive DNS logs and active scans of the IPv4 address space, we construct a methodology for identifying domains whose anomalous network behavior matches the pattern of such attacks and are qualitatively valuable to an attacker.

This work is licensed under a Creative Commons Attribution International 4.0 License.

IMC '22, October 25–27, 2022, Nice, France
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9259-4/22/10.
https://doi.org/10.1145/3517745.3561425

This work makes three contributions:

- *Attack-centric operational signatures.* By identifying the requirements of attack (*i.e.*, obtaining a new certificate, staging a server to host that certificate, using DNS hijacking to divert traffic for a subdomain that handles user credentials to the new server) we construct a model for how such attacks produce network-visible side effects.
- *Opportunistic filtering with existing data sets.* Using a wide array of longitudinal data sets we identify how these side-effects likely manifest in our data. Combined with limited assumptions about attacker behavior, we filter the set of potential victim domains to a modest number.
- *Manual qualitative evaluation.* We evaluate the resulting set of domains manually and qualitatively for their likelihood as potential victims. Our analysis predominantly identifies sensitive government agencies, consistent with the sophisticated mode of attack, and we show that our approach independently identifies virtually all victims documented in the 2019 industry reports.

Ultimately, this paper provides a framework for identifying such attacks as a third party. We explain the complexity in making such identifications but show that existing data, though imperfect, is sufficient to retroactively identify a range of real victims — including sensitive government sites previously undocumented.

2 BACKGROUND

Targeted attackers seek to gain access to an organization and, from there, expand their capabilities. While a small number of such data breaches result directly from exploiting software vulnerabilities (only 3% according to the 2021 Verizon Data Breach report [60]), the vast majority involve the acquisition of remote access credentials (*i.e.*, user names and passwords) typically via phishing, credentials theft and reuse, or brute force. However, all these techniques are fundamentally opportunistic and produce side-effects that can alert system operators (*i.e.*, reported phishing e-mails, failed logins, etc.) An alternative approach is to covertly acquire these credentials in real-time as they are used by remote workers. However, to do this requires the attacker to solve two problems: first, they must be able to divert traffic from remote workers to a server under their control and second, they must bypass any cryptographic protection such as DNSSEC or TLS employed to protect against such diversion.

In this section, we will briefly summarize the relevant aspects of the Domain Name System (DNS), review DNS hijacking in general and DNS infrastructure hijacking in particular, and briefly highlight why existing protections such as DNSSEC and TLS do not protect against these attacks.

2.1 DNS and DNS hijacking

The primary role of the Domain Name System (DNS) is to map human-readable domain names to routable IP addresses for use in a variety of higher-layer protocols and services. DNS provides a hierarchical namespace such that the owner of a given registered domain name (*e.g.*, `nsf.gov`) is delegated authority to resolve that name and any *fully qualified domain names (FQDN)* under it (*e.g.*, `fastlane.nsf.gov`). This works via a query protocol, first specified in RFC 1035 [44], by which any party on the Internet can ask to be directed to an *authoritative nameserver* for a given domain name (its `NS` record). This nameserver then provides the IP address (the `A` record in DNS parlance) that the FQDN maps to.

Any time an attacker can control this resolution process, such that a domain ultimately resolves to an IP address of the attacker's choosing, is referred to as *domain hijacking*. There are a variety of ways that a domain might be hijacked, largely owing to the considerable complexity of the DNS protocol and its implementation. Perhaps the best known is *cache poisoning*, which occurs when an attacker anticipates queries for a domain from a *recursive resolver* (*i.e.*, a service taking responsibility for client DNS resolutions) and injects carefully crafted (but incorrect) responses to satisfy these queries. Because DNS recursive resolvers cache their results, once a domain is poisoned in this way, all the resolver's clients will receive erroneous resolutions [18, 41, 55]. Another class of attack, *query interception*, occurs when the attacker can mediate communications from the requester to its recursive resolver (or from the recursive resolver to other nameservers) and substitute incorrect responses [39, 46, 63]. Yet other attacks involve exploiting configuration errors wherein a domain's `NS` records include so-called *dangling delegations* — nameserver FQDNs whose own domains can be controlled by an attacker (*e.g.*, because they have expired) and then used to influence resolution [3–5, 40].

Finally, *DNS infrastructure hijacks*, which are the focus of this paper, result from an attacker taking control of the mechanism used to update DNS configurations — typically by compromising the domain holder's account with their domain registrar or the systems of the registrar itself.[1] The domain's registrar enjoys privileged access to update the domain's records in its top-level domain (TLD) registry database that is, in turn, relied upon when the domain is resolved. In these attacks, the attacker replaces the `NS` records for the domain with nameservers controlled by the attacker, and arranges that specific `A` records pertaining to targeted subdomains (*e.g.*, `mail`, `vpn`) will point to the attacker's infrastructure. While initial reports of such attacks identified use for activism or mischief [32, 57], more recently several large security firms have reported on their use to compromise government agencies and large infrastructure providers [2, 19, 33, 43, 56].

Thus far there is little academic literature on this issue. One notable exception is Houser *et al.* 's recent paper using a combination of past domain hijacks and Farsight's Passive DNS data set to train a machine learning classifier to detect such hijacks (although the authors do not attempt to use this classifier to detect any new hijacks) [34]. Our work focuses on the same problem domain, but has both different aims and means — we seek to retroactively identify sophisticated attacks in the wild and do so via a constructive framework based on concrete attacker objectives.

[1] Sadly, this is not idle speculation, and we are aware of more than a few instances of registrar compromises. For example, quoting from the indictment of several officers of the Chinese Ministry of State Security *United States v Zhang et al.*: "On August 28, 2013, LIU sent MA a link to a news article that explained how the Syrian Electronic Army (SEA) had hacked into the computer systems of Company L, a domain registrar, in order to facilitate intrusions. On December 3, 2013, members of the conspiracy used the same method as the SEA to hack into the computer systems of Company L and hijack domain names of Company H, which were hosted by Company L." [58]. Similarly, the attacks against `pch.net` (which provides DNS infrastructure services for the ccTLDs of over 130 countries) involved attackers obtaining privileged credentials at `pch.net`'s registrar, Key-Systems [35].

2.2 DNSSEC and TLS

There are multiple standard security mechanisms used to protect against domain hijacking. Specific to DNS, the Domain Name System Security Extensions (DNSSEC) [50] provides cryptographic guarantees of authenticity and integrity for each DNS record, via a trust hierarchy that mirrors the DNS delegation hierarchy. However, DNSSEC is not widely deployed in practice [49] and is of limited benefit in DNS infrastructure hijacks, because it is commonly the very authority for updating DNS records (including their signatures) that has been hijacked [35].

Another defense is provided by the Transport Layer Security (TLS) [48] wherein an application provides certificates signed by a third-party Certificate Authority (CA). These certificates attest that the endpoint represents one or more FQDNs (specified in the Common Name and Subject Alternative Name fields of a TLS certificate) and that the public key contained in the certificate is valid and should be used to bootstrap a secure session. Thus, assuming that clients (*e.g.*, a VPN) validate such certificates before allowing further communication, hijacking a target's DNS resolution will not be sufficient to read the traffic being intercepted. However, the premise underlying this arrangement is that the CA's signed attestation is backed by their appropriate due diligence that the party obtaining the certificate is really who they say they are. However, as we will explain in Section 3, modern certificate provisioning practices have rendered this guarantee itself vulnerable to hijacking.

3 DNS INFRASTRUCTURE HIJACKS

In this section, we describe the stages of the DNS infrastructure hijacks on which we focus, and the challenges in identifying them. DNS infrastructure hijacks are complex and depend on several advanced capabilities.

Develop Capability. Attackers start by developing the ability to modify the target domain's `NS` records or `A` records. This step can leverage three different paths: (a) compromising the account credentials the registrant uses with their registrar or DNS provider; (b) compromising the registrar that administers the domain; or (c) compromising the registry DNS configuration database [2, 35]. In the latter two cases, the hijack can extend to all domains under the registrar's or registry's control. In any of these cases the attacker can also typically disable protections provided by DNSSEC [35].

Attacker Infrastructure. In the previous step the attacker establishes control over a domain's DNS resolution, but the ultimate objective of the attack is to gain control over the *infrastructure* served by the domain. To this end, the adversary must redirect sensitive subdomains — used to receive authentication credentials — to counterfeit infrastructure imitating those services (such as a mail login page), with the goal of accessing credentials via adversary-in-the-middle (AitM) techniques.

Adversary-in-the-Middle Capability. In recent years, a combination of users expecting a "secure connection" and browser vendors initiatives [29–31] make harvesting credentials without a TLS certificate significantly harder.[2] To obtain a TLS certificate that will satisfy modern clients, a domain owner must request one from a browser-trusted *Certificate Authority* (CA) who, in turn, is responsible for verifying domain ownership before provisioning a certificate. While CAs verify ownership using several methods, the one most relevant is *domain validation* which uses demonstrations of real-time control over the domain as a proxy for proof of ownership [21, 26] (*e.g.*, posting a given challenge token in a `TXT` record for the domain). Most certificates today are issued in this manner, using a fully-automated process called the Automatic Certificate Management Environment (ACME) network protocol [1, 7].

Thus, an attacker's ability to control DNS resolution can be sufficient to obtain a browser-trusted TLS certificate for that domain. In fact, there is clear documentation of attackers obtaining certificates for sensitive subdomains (*e.g.*, `mail`, `vpn`) in recent DNS infrastructure hijacks [35, 43]. On obtaining such a certificate, an attacker can then deploy it to its counterfeit infrastructure (*e.g.*, servers for webmail, VPN, IMAP, etc.). Traffic diverted from the target domain to its counterfeit counterpart will accept and use the presented certificate, allowing the attacker to extract the cleartext of any credentials sent. However, acquiring a counterfeit certificate is not entirely covert since the Certificate Transparency (CT) standard [38] requires CA's to publish new certificates in a public audit log before they are issued.[3]

Active Hijack. In this stage the attacker actively redirects resolution of the target domain to their own infrastructure. Typically, this happens serially for short durations over weeks to evade detection [2]. Moreover, if the imitating infrastructure is a reasonable replica of the target and uses a browser-trusted (though maliciously obtained) TLS certificate, users have no immediate way of knowing they have been redirected. Of course, users are more likely to report suspicious activity if they log into the mail service but do not see their mail, so sophisticated attackers tunnel traffic back to the legitimate infrastructure (*e.g.*, using the proxy-like Internet Content Adaption Protocol (ICAP) [15]) to further hide the attack [33]. While traffic is redirected, the attacker collects credentials which can then be used to laterally move into the target organization. While the redirections may last for short durations, the attacker infrastructure typically is functional and responsive throughout.

Post Hijack. Upon successful conclusion of the DNS hijack (*i.e.*, the attacker has acquired login credentials and established a beachhead inside the organization), the attacker can then decommission its counterfeit infrastructure. However, we have seen attackers not only reuse their infrastructure (based on IP addresses observed) for different targets, but also leave infrastructure up for days, sometimes months, after the hijack (Section 5.1).

Summary. This class of DNS hijack is characterized by: (a) multiple brief updates to DNS configuration which minimizes opportunities for detection; (b) attacker infrastructure that is responsive for the duration of the hijack and sometimes much longer; (c) attacker infrastructure that responds with a maliciously obtained browser-trusted certificate.

4 METHODOLOGY

We now describe our methodology for identifying historical DNS infrastructure hijacks. The main insight of our approach is that

[2] Attacks without TLS certificates can still succeed either by socially engineering users to click through security warnings or through the use of drive-by malware [10, 16, 54], but they are not the focus of this paper.

[3] While not strictly mandatory, major browser vendors have made CT participation a pre-requisite for CAs to be trusted by their software.

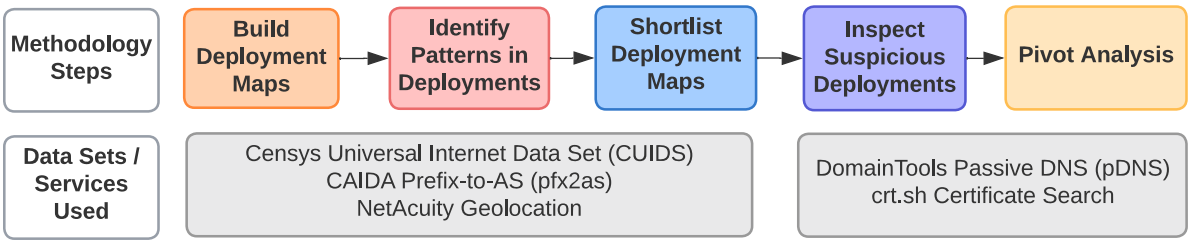

Figure 1: Our five step methodology to identify DNS infrastructure hijacks and the data sets and services used in the steps. Refer to Table 6 in Appendix B for more information on data sets and services used.

Scan Date	IP Address	Ports (TCP)	ASN	CC	crt.sh ID	Issuing CA	Trust	Sens	Name(s) Secured
2019-04-09	84.205.248.69	[443, 993, 995]	35506	GR	1245068498	DigiCert Inc	T	T	`[mail.kyvernisi.gr]`
2019-04-16	84.205.248.69	[443, 993, 995]	35506	GR	1245068498	DigiCert Inc	T	T	`[mail.kyvernisi.gr]`
2019-04-23	95.179.131.225	[993]	20473	NL	1394170951	Let's Encrypt	T	T	`[mail.kyvernisi.gr]`
2019-04-23	84.205.248.69	[443, 993, 995]	35506	GR	1245068498	DigiCert Inc	T	T	`[mail.kyvernisi.gr]`
2019-04-30	84.205.248.69	[443, 993, 995]	35506	GR	1245068498	DigiCert Inc	T	T	`[mail.kyvernisi.gr]`

Table 1: Annotated IP scan data related to `kyvernisi.gr` for the month of April, 2019.

from a third-party vantage point, it is far more feasible to identify the precursor of a hijack — the infrastructure attackers establish to impersonate the domain — rather than tracking changes in DNS configuration that reflect a hijack of the domain. This infrastructure, such as a server that mimics the mail login page of a target domain, has a key requirement for a successful attack: a valid browser-trusted TLS certificate controlled by the attacker to assert authority over the target domain. Thus, once the attacker has established their infrastructure, the certificate will appear in global IPv4 scans of specific TCP ports that return TLS certificates.

Based on this insight, our approach identifies hijacks by first discovering the attacker infrastructure used to target domains. We use data from Internet-wide scans to build a model of deployed infrastructure — a *deployment map* — for every domain over time. We analyze these deployment maps to identify suspicious deployments that strongly correlate with hijacks. Moreover, this approach provides additional context in the form of the longitudinal use of IP addresses, certificates, and CAs associated with each domain. This context provides a baseline for characterizing new infrastructure that appears with authoritative claims on the domain.

Our approach consists of five steps, illustrated in Figure 1 and described in the following sections. We first *build deployment maps* for every domain. Next, we *use patterns* in these maps to *shortlist* domains with potentially suspicious deployments. Then, we *manually inspect* this shortlist with heuristics and supplemental data to establish confidence that the suspicious deployments reveal hijacks. Finally, using the domains inferred as hijacked, we *pivot* to examine if other domains not captured by our methodology were targeted using the same attacker infrastructure.

4.1 Building Deployment Maps

The goal of this stage is to cluster deployments to reveal suspicious infrastructure used to mimic sensitive target domains. In this stage, we take as input *longitudinal* Internet-wide scans that retrieve TLS certificates from responsive hosts to output deployment maps which models *where* and *when* infrastructure on the Internet provided service for a domain. In our work we use the Censys Universal Internet Data Set (CUIDS) [14] with records for more than 71M IP addresses spanning January 2017 to March 2021.

The CUIDS includes comprehensive weekly scans for TLS certificates across the entire IPv4 address space. This weekly scan data captures when a certificate was seen at a specific IP address and port.[4] Starting with this data set, we annotate the IP address with the origin AS (using CAIDA Prefix-to-AS mappings [11]) and its geolocation (using NetAcuity [25]). We further annotate this data with information extracted from the certificate, including the Subject Alternative Names (SANs) [47] specifying the domain names secured by the certificate [27] and the Issuer CA. Additionally, we identify if the certificate is browser-trusted.[5] As an example, Table 1 shows these annotations for four scans that found certificates securing `kyvernisi.gr` in April 2019. Using this annotated data, we identify the observable infrastructure associated with every domain. For this example, scans at two different IP addresses (84.205.248.69 and 95.179.131.225) returned a certificate securing the domain `kyvernisi.gr`. We refer to those IP addresses and the certificates they return as the *observable infrastructure* for `kyvernisi.gr`.

We cluster the observable infrastructure for a domain into separate *deployment groups*. For a given domain, we define a deployment group as the observable infrastructure associated with IP addresses originated by the same ASN on a given date. Our assumption is that the infrastructure used by an attacker will be distinct from the infrastructure used by the domain owner. Thus our goal is to have the legitimate and the attacker infrastructure appear as separate deployment groups in the map.

[4]We use data for ports that are typically associated with TLS certificates and, hence, targeted by attackers (ports [443, 465, 587, 993, 995]).
[5]We mark a certificate as trusted if it is trusted by either Apple, Microsoft, or Mozilla. The Chrome Root Store was rolled out after the time frame considered in our study.

Date	Deployment #1	Deployment #2
2019-04-09	AS35506 [GR] crt.sh_id 1245068498	
2019-04-16	AS35506 [GR] crt.sh_id 1245068498	
2019-04-23	AS35506 [GR] crt.sh_id 1245068498	AS20473 [NL] crt.sh_id 1394170951
2019-04-30	AS35506 [GR] crt.sh_id 1245068498	

Figure 2: Deployment map of `kyvernisi.gr` for April 2019 capturing the two deployments. Deployment #1 is a stable deployment. Deployment #2 is a transient deployment since it only shows up in one scan, indicating suspicious behavior.

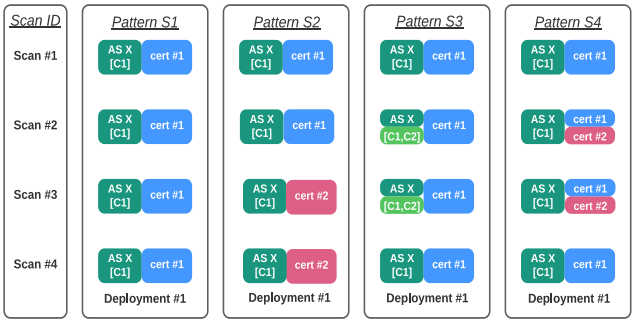

Figure 3: Representative stable patterns (S) in deployment maps. The consistent use of the same ASNs over time indicate stable and benign deployment patterns.

A deployment group seen longitudinally over a period of time is referred to as a *deployment*, and *all* deployments for a domain together represent the domain's *deployment map*. For example, the two rows for the April 23, 2019, scan of `kyvernisi.gr` in Table 1 have IP addresses in two different ASNs (20473 and 35506) that each return certificates for `kyvernisi.gr`. Each forms its own deployment group for that date. As a result, for the period of April 2019 the domain `kyvernisi.gr` has two deployments which, together, form its deployment map as shown in Figure 2.

Instead of building a single deployment map for every domain, we break the period from January 2017 to March 2021 into nine six-month periods.[6] For each of these periods, we build a deployment map for all domains with a publicly-visible TLS certificate in the corresponding six-month period. We consider each period independently, *i.e.*, a domain's lifetime may span multiple deployment maps, each of which we evaluate separately. Breaking up the four-year period leverages temporal locality in deployments to better account for both long-term stable transitions and brief transient changes. From the four years of CUIDS scan data, we construct deployment maps for more than 22M domains.

4.2 Identify Suspicious Patterns

The goal of this stage is to identify attacker infrastructure that appears as new *and* temporary deployments in location and time. In this stage, we take as input the deployment maps built using longitudinal Internet-wide scans to then categorize them into three patterns: *stable*, *transition*, and *transient* deployments. [7] The first two patterns correspond to benign deployments, and the third captures suspicious deployments.

4.2.1 Stable Patterns. The patterns in Figure 3 represent benign, stable deployment maps. Most domains fall into this category: 21.25M (96.5%) of the 22M domains are stable.

Pattern S1 represents a single deployment presenting the same certificate from IP addresses in AS X and geolocated to country $C1$. In this pattern the observed infrastructure for a domain does not change over time, and the certificates associated with the domains have long validity periods. Pattern S2 is similar, but represents the rollover of certificates on expiry in a stable deployment. The only change in observed infrastructure for the domain is a change in the certificate associated with the domain.

Patterns S3 and S4 capture domains that show minor changes in an otherwise stable deployment, such as the appearance of new IP addresses geolocated to a different country but the same AS, or a new certificate securing the domain being deployed on the same observed infrastructure. Such changes could reflect the domain owner testing new endpoints or services, or expanding geographical deployment with the same provider. We consider these patterns as stable and benign given the consistent use of the same AS.

4.2.2 Transition Patterns. While most domains have stable deployments over short time scales, over longer time periods infrastructure associated with domains can change deployments for a variety of legitimate reasons. The patterns in Figure 4 represent transitions that reflect a significant change in observed infrastructure, but the change is stable going forward in time. Such observable transitions in infrastructure will appear as new deployments in the deployment maps. 650K (2.95%) of the domains have transition patterns.

The first two patterns reflect the expansion of domain deployments. Domain owners could be scaling up or diversifying their infrastructure into a new AS (Pattern X1), perhaps even with an additional certificate for use with a new provider (Pattern X2). From our experience examining such deployment maps, these patterns typically correspond to the adoption of cloud services in addition to on-premises infrastructure. The third pattern X3 reflects a shift to completely new infrastructure, with a new certificate for the domain being served from IP addresses in a different AS. A common cause of this pattern is a domain owner switching hosting providers, or the domain changing ownership. At times we see a small overlap between the old and new deployments (the shaded old deployment in the figure). In most cases, DNS resolution switches to the new infrastructure and the previous infrastructure is torn down.

4.2.3 Transient Patterns. The third category of patterns reflects transient changes in deployment maps, and 28K (0.13%) deployment maps fall into this category. Given the transient nature of attacker

[6] We found the six-month period a useful balance between the compute time to build and analyze deployment maps and capturing the typical certificate lifecycle.
[7] We find 77K (0.35%) of domains too noisy or unstable to categorize. Primarily, these domains move deployments continually and have no stable deployment making any inference of hijacking as a third party challenging.

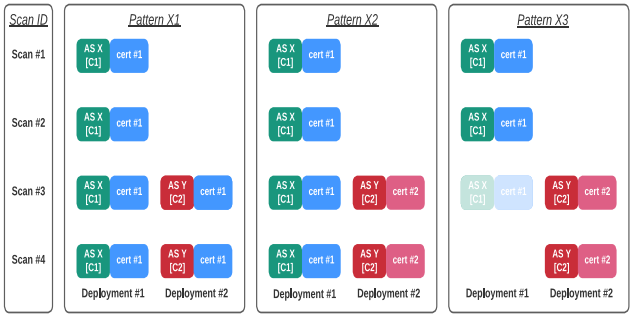

Figure 4: Representative transition patterns (X) in deployment maps. These deployment patterns capture long-term stable changes in deployment.

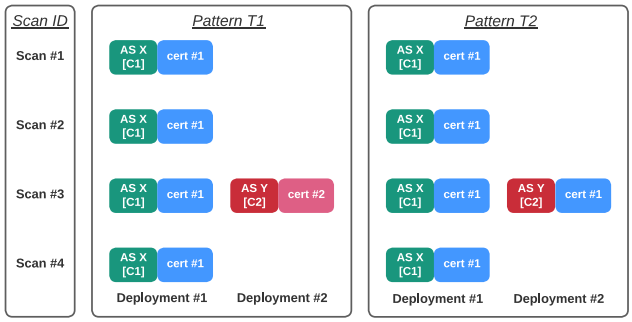

Figure 5: Representative transient patterns (T) in deployment maps. The transient nature of attacker infrastructure mimicking the target domain indicates suspicious deployments.

infrastructure created to mimic the target domain, we expect these patterns to capture such malicious deployments. The key to this identification is the time threshold used to distinguish transient from transition changes. The threshold for attacker infrastructure lifetime needs to be long enough to reflect a malicious deployment, but short enough to avoid false positives. We find three months — the typical validity period of free certificates — to usefully balance these tradeoffs. The intuition is that the attacker infrastructure is tied to the validity of the maliciously-obtained certificate. In most cases during this period, the attacker either harvests credentials to laterally move into the target domain, or the attacker has been discovered. Thus we focus our search on attacks using *transient deployments* that do not persist beyond three months (~12 scans).

Pattern T1 (Figure 5) reflects a deployment map that consists of a long-term stable deployment combined with a transient deployment using a new certificate and different infrastructure. While this suspicious pattern often indicates a hijack, sometimes the evidence is not so conclusive. For instance, a domain may use AS16509 (Amazon) for their stable deployment, but briefly also use AS14618 (also Amazon). It is difficult for a third party to conclude that this activity is malicious — the transient appearance of a different AS for the same provider is not uncommon, but using a new certificate for the domain is. As a result, we label deployment maps matching Pattern T1 as suspicious, and then evaluate them on a case-by-case basis for a final verdict (Section 4.4).

Pattern T2 also reflects the appearance of a transient deployment against the background of a stable deployment, but the certificate associated with the transient deployment is the same as the one used by the stable deployment. This pattern typically indicates a legitimate expansion in infrastructure by the domain owner, but can also reflect malicious activity. In particular, it can capture the prelude to hijacks: the stage of an attack in which the attacker sets up a parallel infrastructure that proxies to the actual IP. Due to the proxy, the certificate scan at an IP address controlled by an attacker returns the legitimate certificates by proxying to an IP address of the stable deployment. Use of the proxy implies that the domain was targeted but not yet hijacked, although it is possible the original certificate was exfiltrated and is being used by the attacker [56]. As with other suspicious deployments, we must manually evaluate deployment maps matching this pattern with care.

4.3 Shortlist Deployment Maps

The goal of this stage is to narrow the candidate set of hijacks. In this stage, we use as inputs the deployment maps identified as transient and then use a set of heuristics to remove common cases that result in false positive or inconclusive results to shortlist the truly suspicious transient deployments.

To shortlist deployment maps, we first check if the transient deployment ASN is organizationally related to the ASN of its stable deployment using the CAIDA AS-to-Organization Mapping [12]. Second, we also check if the transient deployment geolocates to the same countries as any stable deployment. Third, we check if a domain is missing from 20% of scans in the six-month period, or if the domain displays similar transient deployments in multiple (three or more) consecutive six-month periods. These checks identify domains where *our visibility* into the domain's deployment is too unstable to make a determination. We prune deployment maps that match any of these three criteria from our candidate set.

As a final step, we only keep deployment maps where the transient deployment has a browser-trusted certificate securing a *sensitive* subdomain: a subdomain with a substring matching [`secure`, `mail`, `remote`, `login`, `logon`, `portal`, `admin`, `owa`, `vpn`, `connect`, `cloud`, `signin`, `citrix`, `box`, `account`, `intranet`, `imap`, `smtp`, `pop`, `ftp`, `api`]. We manually compiled this list based on common names of subdomains targeted in early attacks. For deployment maps not matching this naming criteria, we still shortlist them if we observe an extremely stable deployment for a six-month period before and after the transient, thus indicating a truly anomalous occurrence.

Thus, our final candidate set contains transient deployments originated by a different ASN *and* geolocated to a different country (relative to the stable deployment) either affecting a sensitive domain or representing a rare anomalous occurrence. The deployment map for `kyvernisi.gr` (Figure 2) is in this set since the transient deployment is for the `mail` subdomain, is originated by a different ASN (AS 20473), and geolocates to a different country (Netherlands) relative to the stable deployment (AS 33506 and Greece). After applying these heuristics, we shortlist 8143 domains as suspicious. Of these, 47 domains are shortlisted for being truly anomalous, i.e., the domain has an otherwise stable deployment for a six-month period before and after the transient deployment.

4.4 Inspect Suspicious Deployments

The goal of this stage is to manually inspect the suspicious transient deployments shortlisted to find corroborating evidence that a hijack has occurred. In this stage, we take as input the shortlisted deployment maps, passive DNS (pDNS) data, and a certificate transparency (CT) data set to evaluate whether a domain has been hijacked. This stage is the most time consuming since it manually examines several features. In this stage we inspect 8143 domains.

We first cross-reference these domains with pDNS and CT data. The main advantage of pDNS data is that it requires no cooperation from zone owners (*e.g.*, it does not require access to restricted zone files for ccTLDs).[8] We use DomainTools' pDNS data set [23]. Since domains targeted for hijacking are by their nature in active use, we expect them to be actively queried. For each domain, pDNS reports the first and last time a specific resolution was seen, if at all. However, pDNS data captures only domains that are actively queried on networks observed by DomainTools' sensors [64]. We also cross-reference these domains with a certificate transparency (CT) data set using the crt.sh [51] service which allows us to search CT logs for certificates issued to a domain. Based on this cross-referencing, we find only 1256 of the 8143 shortlisted domains worth manually examining. For the remaining domains, we neither saw relevant data in the pDNS or CT data in the timeframe around the suspicious transient deployment, nor were they truly anomalous occurrences. In these cases, we found the certificates seen in the transient deployments were typically issued many weeks or months before it became visible. As such, we suspect these cases are legitimate deployments briefly visible to scans making them seem anomalous.

For domains matching Pattern T1 we check pDNS for changes in nameserver delegation and in the resolution of subdomains listed in the suspicious certificate returned from the transient deployment. If the suspicious certificate is *issued* near the time pDNS observes changes in nameserver delegations or changes in domain resolution, then we conclude that the certificate was maliciously obtained and the pDNS changes reflect a hijack. We make this determination given that: the transient deployment is in a different ASN and different country (Section 4.3); it returns a suspicious new certificate for a sensitive domain not used elsewhere; and corroborating data in pDNS records a short-lived change in DNS resolution. We find 22 domains that match these criteria.

For domains matching Pattern T2, confidently inferring that an attack occurred is more challenging, since the transient deployment only captures the prelude to an expected hijack (Section 3). We first check if pDNS captures either a change in DNS resolution for the targeted subdomain or a change in nameserver delegation. We then check the CT logs to see whether a new certificate was issued in the same time period that the transient deployment was observed. If a new certificate exists that secures a sensitive subdomain for which we also observe a change in DNS resolution, we consider the certificate suspiciously obtained.

We conclude a domain was hijacked in the presence of corroborating evidence from both pDNS and CT. We find 6 domains that match this criteria; of those, 4 domains were shortlisted for being truly anomalous occurrences. Notably for `ais.gov.vn`, while we see evidence of redirection in pDNS, we do not find any suspiciously issued certificates. As a result, we mark the domain as *targeted* as opposed to hijacked.

We conclude that a domain has been hijacked only when there is significant corroborating evidence (Section 5.1). In the absence of corroboration, if a domain has a transient deployment that is truly anomalous — it is the only transient deployment in an otherwise stable deployment map — we mark the domain as *targeted but not hijacked*. This situation can arise if the attack was unsuccessful or if none of our data sources captured it. Of the domains shortlisted for being truly anomalous, we find convincing evidence that 24 domains were targeted in this fashion (Section 5.4).

4.5 Pivot Analysis

The goal of this stage is to leverage the domains identified as hijacked to find additional hijacks. In this stage, we use as input the pDNS data and attacker infrastructure used to target domains to find other domains targeted by the same attacker infrastructure. Using the attacker infrastructure of the 28 hijacked domains (22 T1 domains and 6 T2 domains) identified from manual inspection, we pivot to identify other domains referencing the same nameserver delegations or resolving to the same IP addresses using the pDNS logs. This step identifies 13 more domains. We discuss the reasons why deployment maps do not capture these domains in Section 5.2. For these domains, we try to also identify the maliciously obtained certificate using the process detailed above.

4.6 Limitations

Our methodology has several limitations. First, performing longitudinal inference as an independent third party restricts us to publicly available data sets. Thus, as with any other independent third party perspective, our methodology cannot comprehensively identify all hijacks. In absence of groundtruth, delineating an attack from legitimate changes in the extremely dynamic DNS ecosystem relies on multiple data sources to discover corroborating evidence matching the attack profile, and false positives are still a risk. Consequently, we find ways to aggressively prune the data to minimize false positives and inconclusive results (suspicious events with no corroborating data). As a result, our pruning potentially biases our inferences toward domains with more stable standardized deployments for which we could confidently infer were hijacked.

At the same time, we are also limited by coverage issues with the data sets: too coarse-grained to catch ephemeral hijack activity (*e.g.*, weekly active IPv4 scans);[9] addresses that do not respond to scanning; or in the case of traffic data, being limited to those networks where passive DNS traffic is gathered for commercial use (pDNS). Finally, while stages such as building and shortlisting deployment maps are automated, our method eventually requires manual inspection of all candidate hijacks. While our focus on domains that rely on TLS certificates renders this requirement tractable, we hope our experiences enable development of more automated techniques.

Given these limitations, our results are therefore a lower bound (perhaps a severe one). However, our methodology did uncover

[8] The domain hijacks we identified spanned 15 TLDs; of those, CAIDA-DZDB has access to only three of their zone files.

[9] As of April 2021, Censys scans the entire Internet daily, so future studies can overcome this limitation [24].

domains not previously identified, establishing that this class of hijack is a serious ongoing concern.

5 RESULTS

Applying our methodology to historical data between January 2017 and March 2021, we identified 41 domains as hijacked and 24 domains as targeted. To illustrate this approach concretely, we first describe how we use deployment maps to identify a set of related hijacked domains in Kyrgyzstan. We then discuss the overall features of the full set of 41 hijacked domains, independent sources of validation, and the longitudinal implications of the hijacks. We then discuss the features of the 24 domains that were targeted, but for which we did not observe a hijack. Finally we discuss trends in targeted organizations, and the infrastructure used by the attackers.

5.1 The Kyrgyzstan Hijacks

As a concrete example of our method, we describe how we found that a set of Kyrgyzstan government domains were the targets of attacks: `mfa.gov.kg`, `invest.gov.kg`, `fiu.gov.kg`, and `infocom.kg`.

For the four-year duration of our study, the deployment map for `mfa.gov.kg` (the Ministry of Foreign Affairs, Kyrgyzstan) contains a stable infrastructure deployment in Kyrgyzstan, hosted on IP addresses originated by AS 39659 (Infocom, Kyrgyzstan). The deployment map also contains a transient deployment starting December 22, 2020. This transient deployment has a new certificate for the sensitive subdomain `mail.mfa.gov.kg` that is returned from an IP address located in Russia and originated by AS 48282 (VDSINA Hosting, Russia). As a result, the deployment map matches Pattern T1. Additionally, the domain appears in more than 80% of the scans in the six-month period under consideration, the stable and transient deployments are not related to the same AS organization, and are not geolocated in the same countries.

At this point the transient deployment is flagged as suspicious. For a final determination, we use additional data sources for corroborating evidence. From the pDNS data, the stable authoritative nameservers for `mfa.gov.kg` were `ns1.infocom.kg` and `ns2.infocom.kg`. On December 20, 2020, the authoritative nameserver delegations were updated to `ns1.kg-infocom.ru` and `ns2.kg-infocom.ru`, and those nameservers resolved `mail.mfa.gov.kg` to the IP address using the suspicious certificate. Both NS and A redirections continued until January 12, 2021. Using the CT logs from `crt.sh`, the certificate returned from the transient deployment for `mail.mfa.gov.kg` was issued on December 21, 2020, by Let's Encrypt.[10]

Given all of the evidence — a transient deployment in a different AS that returns a new, suspicious certificate for `mail.mfa.gov.kg`, together with changes to the authoritative nameservers at the same time that the new certificate was issued — we conclude that: the domain was hijacked, and the attacker infrastructure used IP address 94.103.91.159 and nameservers `ns{1,2}.kg-infocom.ru`.

The domain `invest.gov.kg` (Investment Portal, Kyrgyzstan) was attacked on December 28, 2020, a week later than `mfa.gov.kg`. The domain `invest.gov.kg` also has a transient deployment with a new certificate for `mail.invest.gov.kg`, and it uses the same anomalous AS and authoritative nameservers as the attack on `mfa.gov.kg`. These nameservers redirected `mail.invest.gov.kg` to the transient deployment on December 28, 2020, for a week.

After identifying that `mfa.gov.kg` and `invest.gov.kg` were attacked, we pivot to investigate whether other domains were targeted using the same attacker network and nameserver infrastructure. From the pDNS data, we see `ns{1,2}.kg-infocom.ru` being briefly used as authoritative nameservers for `fiu.gov.kg` (Financial Intelligence Service, Kyrgyzstan) in December 2020 and for `infocom.kg` (State Agency for Information Services) in January 2021. Those anomalous nameservers returned resolutions for `mail.fiu.gov.kg` and `mail.infocom.kg` to a server in the same AS as the attacker infrastructure for the other two domains.[11] Cross-referencing with the CT logs at `crt.sh` shows new TLS certificates issued for `mail.fiu.gov.kg` and `mail.infocom.kg` in the same time frame. Based upon this evidence, we also conclude that the domains `fiu.gov.kg` and `infocom.kg` were the targets of a hijack.

The deployment maps for `fiu.gov.kg` and `infocom.kg` did not match a transient pattern because the IP scans did not find any stable observable infrastructure for the domains. This case demonstrates the utility of the pivot step, enabling discovery of attacks for domains that do not have stable observable infrastructure.[12]

5.2 Hijacked Domains

Table 2 reports the 41 domains we infer as hijacked. These domains span government agencies, infrastructure providers, and even registrar and registry operators. Since most of the domains are associated with government agencies, we group the domains by their country (CC) and order each group by the time of hijack (Hij). For each domain, we report the reason we identify the domain (Type) and the subdomain targeted (Sub). We also report whether there is corroborating nameserver (pDNS) or certificate transparency (CT) evidence, as well as the network infrastructure and location of the victim and attacker deployments.

We identify 20 domains as hijacked directly using deployment maps, indicated by the pattern T1. As with the Kyrgyzstan domains discussed above, the deployment maps revealed: a transient deployment in a different AS and country; the transient deployment returned a suspicious newly-issued certificate targeting one specific sensitive subdomain; and resolutions in pDNS revealed short-lived changes to the authoritative nameservers that briefly redirected traffic to the infrastructure (IP address) in the transient deployment.

We identify another 6 domains as hijacked using a combination of deployment maps and CT data, indicated by the pattern T2. For these cases, we first captured the prelude to the hijack. While we see a transient deployment in a different AS and country, the transient deployment returns a certificate associated with the stable deployment. However, on further inspection the pDNS logs revealed short-lived changes in resolution for a sensitive subdomain from the stable deployment to the transient deployment. Cross-referencing the subdomain with the CT logs reveals a suspicious newly-issued certificate for the sensitive subdomain in the same time frame:

[10] https://crt.sh/?id=3810274168

[11] The legitimate nameservers for these domains are under `infocom.kg`.

[12] While these hijacks focus on harvesting credentials, `mfa.gov.kg` was redirected again in May 2021 (outside the period of our study) luring users into installing a "security update". This executable links to a malware family known as Tomiris which researchers have linked to SolarWinds [37]. Appendix A briefly discusses this finding.

Type	Hij.	Targeted Domain Information			Cross Ref		Attacker Infra. (Transient)			Legitimate Infra. (Stable)	
		CC	Domain	Sub.	pDNS	crt	IP	ASN	CC	ASNs	CCs
T1	May'18	AE	mofa.gov.ae	webmail	✔	✔	146.185.143.158	14061	NL	[5384,202024]	[AE]
T1	Sep'18	AE	adpolice.gov.ae	advpn	✔	✔	185.20.187.8	50673	NL	[5384]	[AE]
T1*	Sep'18	AE	apc.gov.ae	mail	✖	✔	185.20.187.8	50673	NL	[5384]	[AE]
T2	Sep'18	AE	mgov.ae	mail	✔	✔	185.20.187.8	50673	NL	[202024]	[AE]
T1	Jan'18	AL	e-albania.al	owa	✔	✔	185.15.247.140	24961	DE	[5576]	[AL]
T2	Nov'18	AL	asp.gov.al	mail	✔	✔	199.247.3.191	20473	DE	[201524]	[AL]
T1	Nov'18	AL	shish.gov.al	mail	✔	✔	37.139.11.155	14061	NL	[5576]	[AL]
T1	Dec'18	CY	govcloud.gov.cy	personal	✔	✔	178.62.218.244	14061	NL	[50233]	[CY]
P-IP	Dec'18	CY	owa.gov.cy	.	✔	✔	178.62.218.244	14061	NL	[50233]	[CY]
T1	Dec'18	CY	webmail.gov.cy	.	✔	✔	178.62.218.244	14061	NL	[50233]	[CY]
P-IP	Jan'19	CY	cyta.com.cy	mbox	✔	✔	178.62.218.244	14061	NL	—	—
T1	Jan'19	CY	sslvpn.gov.cy	.	✔	✔	178.62.218.244	14061	NL	[50233]	[CY]
T1	Feb'19	CY	defa.com.cy	mail	✔	✔	108.61.123.149	20473	FR	[35432]	[CY]
T1	Nov'18	EG	mfa.gov.eg	mail	✔	✔	188.166.119.57	14061	NL	[37066]	[EG]
T2	Nov'18	EG	mod.gov.eg	mail	✔	✔	188.166.119.57	14061	NL	[25576]	[EG]
T2	Nov'18	EG	nmi.gov.eg	mail	✔	✔	188.166.119.57	14061	NL	[31065]	[EG]
T1	Nov'18	EG	petroleum.gov.eg	mail	✔	✔	206.221.184.133	20473	US	[24835,37191]	[EG]
T1	Apr'19	GR	kyvernisi.gr	mail	✔	✔	95.179.131.225	20473	NL	[35506]	[GR]
T1	Apr'19	GR	mfa.gr	pop3	✔	✔	95.179.131.225	20473	NL	[35506,6799]	[GR]
T2	Sep'18	IQ	mofa.gov.iq	mail	✔	✔	82.196.9.10	14061	NL	[50710]	[IQ]
P-IP	Nov'18	IQ	inc-vrdl.iq	.	✔	✔	199.247.3.191	20473	DE	[50710]	[IQ]
P-NS	Dec'18	JO	gid.gov.jo	.	✔	✔	139.162.144.139	63949	DE	—	—
P-NS	Dec'20	KG	fiu.gov.kg	mail	✔	✔	178.20.41.140	48282	RU	—	—
T1	Dec'20	KG	invest.gov.kg	mail	✔	✔	94.103.90.182	48282	RU	[39659]	[KG]
T1	Dec'20	KG	mfa.gov.kg	mail	✔	✔	94.103.91.159	48282	RU	[39659]	[KG]
P-NS	Jan'21	KG	infocom.kg	mail	✔	✔	195.2.84.10	48282	RU	—	—
T1	Dec'17	KW	csb.gov.kw	mail	✔	✔	82.102.14.232	20860	GB	[6412]	[KW]
P-IP	Dec'18	KW	dgca.gov.kw	mail	✔	✔	185.15.247.140	24961	DE	—	—
T1*	Apr'19	KW	moh.gov.kw	webmail	✖	✔	91.132.139.200	9009	AT	[21050]	[KW]
T2	May'19	KW	kotc.com.kw	mail2010	✔	✔	91.132.139.200	9009	US	[57719]	[KW]
P-IP	Nov'18	LB	finance.gov.lb	webmail	✔	✔	185.20.187.8	50673	NL	—	—
P-IP	Nov'18	LB	mea.com.lb	memail	✔	✔	185.20.187.8	50673	NL	—	—
T1	Nov'18	LB	medgulf.com.lb	mail	✔	✔	185.161.209.147	50673	NL	[31126]	[LB]
T1	Nov'18	LB	pcm.gov.lb	mail1	✔	✔	185.20.187.8	50673	NL	[51167]	[DE]
P-IP	Oct'18	LY	embassy.ly	.	✔	✖	188.166.119.57	14061	NL	—	—
P-NS	Oct'18	LY	foreign.ly	.	✔	✔	188.166.119.57	14061	NL	—	—
T1	Oct'18	LY	noc.ly	mail	✔	✔	188.166.119.57	14061	NL	[37284]	[LY]
T1	Jan'18	NL	ocom.com	connect	✔	✔	147.75.205.145	54825	US	[60781]	[NL]
P-NS	Jan'19	SE	netnod.se	dnsnodeapi	✔	✔	139.59.134.216	14061	DE	—	—
T1	Mar'19	SY	syriatel.sy	mail	✔	✔	45.77.137.65	20473	NL	[29256]	[SY]
P-NS	Dec'18	US	pch.net	keriomail	✔	✔	159.89.101.204	14061	DE	—	—

Table 2: List of 41 domains identified as hijacked between January 2017 and March 2021. Type indicates how we identified the domain. For every domain, we report the time of first hijack, the targeted subdomain, the country associated with the organization behind the domain, corroborating evidence from pDNS and CT, as well the network infrastructure and geolocation of the attacker and the target domain. The 33 domains not highlighted are associated with Sea Turtle campaigns. The domains highlighted in gray have not previously been identified. The .kg domains partially match findings in a report by Kaspersky [37].

although the transient deployment did not return the suspicious certificate in the CUIDS scans, it did appear in the CT logs.

Then we use the attacker infrastructure to pivot, identifying another 13 domains as hijacked. We identify 6 domains pivoting on the attacker infrastructure IP address (indicated as P-IP), and another 7 domains pivoting on the attacker-controlled authoritative nameservers (indicated as P-NS). For all but one, we also find corroborating evidence in the CT logs in the form of suspicious newly-issued certificates for the sensitive subdomains in the same time frame.

These domains are not directly flagged using deployment maps for a couple of reasons. First, there might not be any observable infrastructure for the domain revealed using the IP address scans (*e.g.*, `dgca.gov.kw`), or a domain might not use a TLS certificate (*e.g.*, `embassy.ly`). Second, some deployment maps have many deployments, making it challenging to conclude which correspond to a suspicious transient deployment; both `netnod.se` and `owa.gov.cy`, for instance, have multiple transient deployments.

The final two domains, `apc.gov.ae` and `moh.gov.kw`, do not have corroborating pDNS evidence (indicated as T1*). However, we identify them as hijacked because we see a transient deployment with a suspicious newly-issued certificate securing a sensitive subdomain *and* the exact IP address associated with the transient deployment was also used to hijack other domains.

Validation. As discussed in Section 3, a key challenge with a third-party approach to identifying these attacks is the lack of ground truth. The goal of these attacks is to compromise an organization, and organizations are typically reluctant to publicize such events when they happen.

However, there are multiple reasons to believe that the attacks we have identified are authentic. First, our methodology is not a machine learning approach subject to overfitting — there is no training or training data. Our methodology is to identify the critical operational requirements for this class of attacks. Thus, while we may miss attacks for lack of data, all real attacks that appear in our data set should be identified by our approach. Second, in spite of our constructive approach, the sites our methodology identified are almost exclusively *government agencies* — precisely the sites that we would expect to be targeted in sophisticated attacks.

Finally, in addition to these circumstantial observations, many sites we identified have been independently confirmed as targets of DNS hijacking attacks — either directly (*i.e.*, the site itself is named) or indirectly (*i.e.*, attacker infrastructure IPs are named). Thus, for every hijacked domain and attacker IP address from Table 2, we searched online for articles that include either feature.

Notably, the attacker infrastructure targeting 33 domains matched the IP addresses implicated in the Sea Turtle hijacks [2, 19]. Additional articles confirm 28 of these 33 domains as being targets of DNS hijacks [35, 52]. Further, a recent report by Kaspersky partially matches our findings about the four `.kg` domains [37].

Four domains (highlighted in gray) do not have independent confirmations. We continue to believe that DNS hijacking is the most likely explanation, even if these attacks have not been previously discovered or publicly disclosed.

Longitudinal Patterns. The hijacks span the entire four years of our data set, with a significant uptick in 2018 corresponding with the Sea Turtle hijacks. Perhaps more significant, we find recurring hijacks of domains under the same TLD spanning months and in some cases years. This pattern suggests that the attackers were sophisticated enough to evade detection for long periods. Moreover, we identify domain hijacks as late as January 2021, long after the Sea Turtle hijacks were publicized (early 2019), indicating that these types of attacks remain an ongoing problem.

5.3 Observability

Attackers are very careful in limiting the durations of domain hijacks to minimize observability via DNS. When attackers change the domain resolutions to their infrastructure, they generally do so for less than a day at a time. This approach prevents the hijack from not only appearing in the daily zone file snapshots, even if the domain is hijacked more than once, but it also avoids triggering alerts at the victim organization due to prolonged reduction in traffic. For 51% of the domains hijacked, pDNS captures evidence of the attack itself — domain resolutions to the malicious infrastructure — for at most one day. Of the three domains whose TLDs we have zone file access to, the hijack is not visible in the zone file for two of the domains (`ocom.com` and `netnod.se`). For the third, `pch.net`, the hijack is visible in the zone for one day — yet resolutions to the malicious infrastructure appear in pDNS spanning a 20-day period.

Attacker infrastructure is observable for longer periods. Attackers create the malicious certificate and deploy it relatively quickly. More than 50% of the malicious certificates for the hijacked domains are visible in the certificate scans within 8 days of the certificate being issued and appearing in the certificate transparency logs. Once deployed, though, the malicious certificate is often observable in only a small number of weekly CUIDS scans. For more than 50% of the domains, the malicious certificate only appeared in one scan, and another 20% of the certificates appeared in just two.

5.4 Targeted Domains

Recall that the second pattern of transient deployments (T2) captures the prelude to the hijack. The deployment maps reveal a transient deployment in an unrelated AS located in a different country relative to a long-term stable deployment. However, the transient deployment returns the same certificate as the stable deployment, and there is no evidence that the authoritative nameservers are changed to use the transient deployment. Our interpretation of these deployment maps is that they either correspond to attacks that never launched, our data sets failed to capture the relevant events, or that, while highly anomalous, they reflect legitimate activity that we cannot discern from a third-party perspective.

Table 3 lists the 24 domains we identify as targeted. Similar to the hijacked domains, we group the domains by their country and order them by the time of hijack. Many of these domains show attacker infrastructure being reused. For instance, attacker infrastructure targeting four domains across two ccTLDs (`.ae`, `.sa`) uses the same IP address (194.152.42.16). The same IP address (103.213.244.205) is also used to target two `.vn` domains with two distinct stable deployments. Moreover, attackers use AS 45102 (Alibaba) to target domains across eight TLDs (`.ch`, `.kz`, `.lt`, `.lv` `.ma`, `.mm`, `.gov`, `.vn`) between June 2020 and November 2020.

We also note that 21 of the 24 domains were targeted in 2020, after the Sea Turtle disclosures, so perhaps this activity reflects a

Tar. Date	Targeted Domain CC	Domain	Sub	Cross Ref. pDNS	crt	Attacker Infra. (Transient) IP	ASN	CC	Legit. Infra. (Stable) ASNs	CCs
Apr'20	AE	milmail.ae	—	✗	✗	194.152.42.16	47220	RO	[5384]	[AE]
Apr'20	AE	mocaf.gov.ae	—	✗	✗	194.152.42.16	47220	RO	[5384]	[AE]
Apr'20	AE	moi.gov.ae	—	✗	✗	194.152.42.16	47220	RO	[5384]	[AE]
Dec'20	AE	epg.gov.ae	—	✗	✗	159.69.193.152	24940	DE	[202024]	[AE]
Jun'20	CH	parlament.ch	—	✗	✗	8.210.146.182	45102	SG	[61098,3303]	[CH]
Nov'20	GH	nita.gov.gh	—	✗	✗	78.141.218.158	20473	NL	[37313]	[GH]
Sep'17	JO	psd.gov.jo	mail	✗	✗	185.162.235.106	50673	NL	[8934]	[JO]
Jun'20	KZ	zerde.gov.kz	—	✗	✗	8.210.190.81	45102	SG	[48716,15549]	[KZ]
Nov'20	LT	stat.gov.lt	—	✗	✗	8.210.190.214	45102	SG	[6769]	[LT]
Jul'20	LV	iem.gov.lv	—	✗	✗	8.210.199.85	45102	SG	[8194, 25241]	[LV]
Nov'20	LV	zva.gov.lv	—	✗	✗	8.210.36.66	45102	SG	[8194, 199300]	[LV]
Apr'18	MA	justice.gov.ma	micj	✓	✗	188.166.160.110	14061	DE	[6713]	[MA]
Apr'20	MA	mem.gov.ma	—	✗	✗	47.75.34.153	45102	HK	[6713]	[MA]
Oct'20	MM	mofa.gov.mm	—	✗	✗	47.242.150.18	45102	US	[136465]	[MM]
Nov'20	PL	knf.gov.pl	—	✗	✗	103.195.6.231	64022	HK	[34986]	[PL]
May'20	SA	cmail.sa	—	✗	✗	194.152.42.16	47220	RO	[49474]	[SA]
Sep'20	TM	turkmenpost.gov.tm	—	✗	✗	185.229.225.228	41436	NL	[20661]	[TM]
Aug'20	US	manchesternh.gov	—	✗	✗	8.210.210.235	45102	SG	[13977]	[US]
Dec'20	US	batesvillearkansas.gov	host	✗	✗	95.179.153.176	20473	NL	[32244]	[US]
Apr'19	VN	ais.gov.vn	intranet	✓	✗	45.77.45.193	20473	SG	[131375,63748]	[VN]
Dec'20	VN	mofa.gov.vn	—	✗	✗	45.77.27.9	20473	JP	[24035]	[VN]
Mar'20	VN	cpt.gov.vn	—	✗	✗	103.213.244.205	136574	JP	[63747]	[VN]
Mar'20	VN	most.gov.vn	—	✗	✗	103.213.244.205	136574	JP	[38731,131373]	[VN]
Sep'20	VN	vass.gov.vn	—	✗	✗	47.74.3.121	45102	JP	[18403]	[VN]

Table 3: List of 24 Domains identified as targeted for hijacking between January 2017 and March 2021. The deployment maps for all these domain match Pattern T2 which is typically a prelude to the actual attack. These domains represent truly anomalous occurrences — an IP from another ASN from another country returned a certificate for the domain. Similar to the hijacked domains, we report on both the inferred victim and attacker infrastructure.

completely different set of attackers. Unfortunately, we do not have a way of making that determination, nor do we find any online reports mentioning either these domains or the IP addresses of the transient deployments.

5.5 Affected Organizations

We manually identified the organizations associated with the domains identified as hijacked or targeted (Tables 7 and 8 in Appendix B). These organizations span government ministries, government organizations, infrastructure providers, and even some private firms. Table 4 breaks down the affected organizations by sector, listing the number of domains identified as hijacked or targeted. Domains associated with governments top this list, suggesting state-affiliated motivations behind the attackers and the style of their attacks. Domains related to government Internet services (mail, cloud, VPN services) reflect their value for credential theft.

Organization Sector	# of Domains Hij.	Tar.	Total
Government Ministry	12	11	23
Government Organization	4	6	10
Government Internet Services	7	0	7
Infrastructure Provider	6	0	6
Law Enforcement	3	1	4
Energy Company	3	0	3
Intelligence Services	3	0	3
Postal Service	0	3	3
Civil Aviation	2	0	2
Local Government	0	2	2
Insurance	1	0	1
IT Firm	0	1	1
Total	41	24	65

Table 4: Affected organizations by sector.

5.6 Attacker Infrastructure

As a final analysis we examine features of the attacker infrastructure used to hijack or target domains.

Network Information		# of Domains		
ASN	AS Name	Hij.	Tar.	Total
14061	Digital Ocean	15	1	16
20473	Vultr	7	4	11
45102	Alibaba	0	9	9
50673	Serverius	7	1	8
48282	VDSINA	4	0	4
47220	ANTENA3	0	4	4
9009	M247	2	0	2
24961	MYLOC	2	0	2
63949	Linode	2	0	2
136574	Zheye Network	0	2	2
20860	IOMart	1	0	1
54825	Packet Host	1	0	1
24940	Hetzner	0	1	1
41436	CloudWebManage	0	1	1
64022	Kamatera	0	1	1
	Total	41	24	65

Table 5: Networks used by attackers.

Network. Table 5 lists the networks used by attackers and the number of domains targeted. It shows a concentration in the use of Digital Ocean, Vultr, and Serverius. While we see concentrations in networks used, we do not believe they are reliable features for detection since attacker infrastructure is largely portable. As one example, for the domain `owa.gov.cy` the attacker targeted the domain from four separate ASNs: 14061 , 20473, 33387, 44901. We also see a difference in the ASes used between hijacked and targeted domains, which likely simply reflects different attackers being observed.

Certificates. Another important aspect of the attacker infrastructure are the certificates. Of the hijacked domains, we identified a suspicious certificate for 40 domains. Table 9 in Appendix B lists these certificates along with the CAs which issued them. These certificates were issued from two CAs: 28 from Let's Encrypt, and 12 from Comodo.[13] Let's Encrypt offers free, automated certificate issuance via its ACME protocol [7], and Comodo (now Sectigo) offers free trial certificates [6]. Both use Domain Validated (DV) certificates which only requires control over DNS infrastructure [22]. Given that other CAs (*e.g.*, ZeroSSL) now also offer automated certificates, it would not be surprising to see other CAs being used by attackers going forward. Significantly, only four of these certificates were revoked based on the Certificate Revocation List (CRL) indexed by `crt.sh` as provided by the issuer CA.[14] This lack of revocation suggests that, in most of the cases, the victim is unaware of the hijack until after the certificate expiry, if at all.

As an interesting data point, we found that the legitimate infrastructure of some domains used certificates which were not browser-trusted, indicating use of an internal trusted CA by the domain owner. This use of an internal trusted CA means that the CT logs only contain the suspicious certificates associated with a transient deployment.

[13]Comodo has been since rebranded to Sectigo. Two domains issued by Sectigo are counted as issued by Comodo.

[14]Let's Encrypt does not provide a CRL for the leaf certificates and instead relies on the Online Certificate Status Protocol (OCSP) [17]. As a result, we cannot determine if the certificates issued by Let's Encrypt were revoked.

6 ETHICS

In this paper, we identified a number of historical hijacks that were previously unidentified. While the direct harms surrounding these events have long since past, we did not know if the victims were aware of these incidents (and hence able to make appropriate decisions concerning their security moving forwards such as resetting passwords, etc.). Thus, we believe that victim notification was our primary ethical obligation. We reached out to the previously unidentified 8 hijacked domains and 24 targeted domains, directly and via their national CERTs and reported all domains and inferred attacker infrastructure to allow for full auditing. Over five months have passed since our notifications so we now believe all affected parties are well aware of these potential issues. While we recognize that our publication of this data also creates the potential for secondary reputational harms (governments, in particular, do not like to be seen as victims — perhaps explaining the lack of responses to our notifications) we believe those interests are secondary to the value of full transparency for the broader research and security communities. Indeed, just as we have benefited from detailed third-party reporting to help evaluate our own research, we believe our data can and will provide purchase for other researchers to further investigate strategies used by attackers (*e.g.*, Appendix A) and to provide examples for reasoning about detection and prevention.

Finally, while the hosting providers used by attackers were not, themselves, victims, we have recently initiated outreach to this community (notably to Digital Ocean which was highly represented in our data set) in the hope that they may be able to provide additional insight concerning such attacks. As of yet, it is unclear if this pursuit will be fruitful.

7 DISCUSSION

The key result of this work is a methodology for retroactively identifying evidence of targeted DNS infrastructure hijacking. We identify a range of potential victims (predominantly government agencies) in over twenty countries, including many that have been independently confirmed via forensic reporting but also a variety that have never been reported publicly. Here we reflect on the lessons learned from identifying such attacks, potential future directions, and the work required to mitigate and improve visibility into such threats.

7.1 Lessons from Retroactive Identification

Perhaps the most remarkable result from this study is that it was possible at all. The fundamental challenge of this study was to confidently infer that changes to a domain's deployment represented a hijack as opposed to a legitimate change. Crucial to this inference was an integrated approach using data from multiple sources to provide corroborating evidence. That said, our approach is fundamentally limited by our third-party view point. As such, absent ground truth, we have no way to judge the comprehensiveness or representativeness of our results. Moreover, given our aggressive pruning, it is entirely likely that our methodology fails to identify

a range of more complex or subtle hijacks. What our study demonstrates that it is *possible* to identify hijacks as a third party, but the limit of this approach remains an open question.

Indeed, the success of this work suggests two potential future directions: improving retroactive identification, and exploring interventions that impede attacker workflows. The first direction is to improve our methodology by relaxing our constraints and incorporating additional information (*e.g.*, changes in DNSSEC status during the time-frame of a transient deployment, or the source IP used to request or verify a certificate from its CA). Evaluation remains an open challenge however and in the absence of ground truth it seems likely that retroactive identification will still require significant manual evaluation.

The second opportunity is to identify potential interventions that can actively impede the attacker's workflow in targeted hijacks. However, most such interventions hinge on the ability to detect hijacks in near real-time. One possibility worth exploring is automatically triggering reactive DNS measurements on certificate issuance. Such reactive DNS measurement data could then be cross-referenced with historical deployment maps to flag suspicious certificate issuance. Using follow-on reactive measurements, one might then infer a hijack by identifying when changes to nameserver delegations were transient.

7.2 Mitigating Infrastructure Hijacks

With sophisticated attackers targeting the weakest link in the chain to compromise security, mitigating infrastructure hijacks effectively requires systematic attention across many disparate entities — registrars, registries, CAs, DNS service providers, ISPs, software developers, and site operators. No single party is in an ideal position to address this problem completely and effectively. Moreover, the overheads for mitigation will frequently accrue to those other than the beneficiaries. The implicit trust between the entities means that, while organizations can choose to use mechanisms such as DNSSEC, TLS, or two factor authentication (2FA) for the registrant account, an attacker who can compromise a registrar, registry, or a DNS provider can effectively bypass these mechanisms. More generally, defenses at any single entity are conditional on the defenses of the entities upstream. As such, currently the most practical recourse for most organizations is to constantly monitor their own DNS configuration. To enable this monitoring by all parties, the need for transparency in DNS at short time scales is important.

Transparency. Unlike other ecosystems such as routing (BGP), the DNS ecosystem (especially ccTLDs) is comparatively opaque at short time scales. Because the goal of these attacks is to gather credentials, they only need to be active for extremely short periods of time — once to acquire a certificate and again to harvest credentials. Indeed, with vanishingly few exceptions, these attacks are entirely invisible in DNS zone files because their daily granularity is orders of magnitude too coarse to capture the attack. Similarly, both active and passive DNS measurements only record such attacks if they happen to measure the DNS state at *precisely* the time that a hijack is taking place. That brief anomalies — both benign and malicious — are largely invisible to existing measurement infrastructure is well understood by sophisticated adversaries. Given monitoring is an important tool to mitigate hijacks, this lack of transparency in DNS is an important challenge for our community — whether through online change detection, reactive measurement, or systems (such as CT logs) that log all potential state-impacting changes.

Implicit Trust Dependence. Another issue is the ongoing challenge of implicit trust dependence. While TLS is designed to protect us from actors mounting adversary-in-the-middle attacks, its security depends on the due diligence of trusted CAs. Yet the economic efficiencies of CAs using domain validation has produced an environment where a DNS infrastructure hijack is sufficient to subvert this due diligence and thus bypass TLS — an authentication ouroboros. Similarly, protocols like DNS and DNSSEC implicitly place trust in registrars and registries. But if one is compromised, the guarantees made by these protocols are easily bypassed. This is not unique to the situation described in this paper, and there are a plethora of well-documented attacks around trust dependency issues ranging from BGP and DNS [9, 53] to package managers [45]. This remains an open area of research, but virtually all solutions take some page from the "trust but verify" book. In much the same way that Let's Encrypt now guards against BGP hijack [9], we will need to develop similar capabilities against DNS hijack and improved Registrar and Registry Lock features [28, 36, 59].

Cleartext Credentials. Given the targeting of credentials, we may want to rethink the practice of sending cleartext user credentials. While violations of TLS' integrity and confidentiality guarantees are problematic, that a single violation should provide long-term arbitrary access inside a target organization is an asymmetric threat. The independent efforts to transition to password-less authentication, via WebAuthn, CTAP and its successors, may provide an opportunity to eliminate password theft as a potential attack vector for Internet-facing services.

Overall, addressing this problem effectively will, in addition to further research and development, require significant investment in leadership and coordination across the various stakeholders.

ACKNOWLEDGMENTS

We thank our shepherd John Heidemann and the reviewers for their insightful and constructive suggestions and feedback. This work has benefited from the help of many individuals and organizations. At UCSD we thank Cindy Moore, Bradley Huffaker, and Daniel Anderson for supporting the software and hardware infrastructure for this project, and we thank Daniel Park for his legal guidance. We are indebted to Ian Foster for his help with CAIDA-DZDB/DNS Coffee, and also thank Benjamin Braun, Louis Dekoven, Shuai Hao, and Bill Woodcock for their help and insights over time. We also appreciate the thoughtful help and feedback we have received from many sectors of the community. We thank Sean McNee, Jackie Abrams, Ariella Robinson, and Susan Prosser from DomainTools for their help with pDNS data. We are also grateful to Jacob Hoffman-Andrews, James Renken, and Samantha Frank from Let's Encrypt for sharing their insights on hijacking from a CA perspective. We are also thankful to Levi Richardson, Joe St Sauver, Colin Walker and Eric Ziegast for their help. This work was supported in part by National Science Foundation grants OAC-2131987, CNS-1705050, and CNS-2152644, the EU H2020 CONCORDIA project (830927), the NWO-DHS MADDVIPR project (628.001.031/FA8750-19-2-0004), and generous support from DigiCert.

REFERENCES

[1] Josh Aas, Richard Barnes, Benton Case, Zakir Durumeric, Peter Eckersley, Alan Flores-López, J. Alex Halderman, Jacob Hoffman-Andrews, James Kasten, Eric Rescorla, Seth Schoen, and Brad Warren. 2019. Let's Encrypt: An Automated Certificate Authority to Encrypt the Entire Web. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (London, United Kingdom) *(CCS '19)*. Association for Computing Machinery, New York, NY, USA, 2473–2487. https://doi.org/10.1145/3319535.3363192

[2] Danny Adamitis, David Maynor, Warren Mercer, Matthew Olney, and Paul Rascagneres. 2019. DNS Hijacking Abuses Trust In Core Internet Service. https://blog.talosintelligence.com/2019/04/seaturtle.html.

[3] Gautam Akiwate, Mattijs Jonker, Raffaele Sommese, Ian Foster, Geoffrey M. Voelker, Stefan Savage, and KC Claffy. 2020. Unresolved Issues: Prevalence, Persistence, and Perils of Lame Delegations. In *Proceedings of the ACM Internet Measurement Conference* (Virtual Event, USA) *(IMC '20)*. Association for Computing Machinery, New York, NY, USA, 281–294. https://doi.org/10.1145/3419394.3423623

[4] Gautam Akiwate, Stefan Savage, Geoffrey M. Voelker, and K C Claffy. 2021. Risky BIZness: Risks Derived from Registrar Name Management. In *Proceedings of the 21st ACM Internet Measurement Conference* (Virtual Event) *(IMC '21)*. Association for Computing Machinery, New York, NY, USA, 673–686. https://doi.org/10.1145/3487552.3487816

[5] Eihal Alowaisheq, Siyuan Tang, Zhihao Wang, Fatemah Alharbi, Xiaojing Liao, and XiaoFeng Wang. 2020. Zombie Awakening: Stealthy Hijacking of Active Domains through DNS Hosting Referral. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security* (Virtual Event, USA) *(CCS '20)*. Association for Computing Machinery, New York, NY, USA, 1307–1322. https://doi.org/10.1145/3372297.3417864

[6] Comodo Certification Authority. 2022. Comodo SSL Single DV Certificate. https://ssl.comodo.com/comodo-ssl-dv-trial.

[7] Richard Barnes, Jacob Hoffman-Andrews, Daniel McCarney, and James Kasten. 2019. Automatic Certificate Management Environment (ACME). RFC 8555. https://doi.org/10.17487/RFC8555 https://www.rfc-editor.org/info/rfc8555.

[8] Tara Siegel Bernard, Tiffany Hsu, Nicole Perlroth, and Ron Lieber. 2017. Equifax Says Cyberattack May Have Affected 143 Million in the U.S. https://www.nytimes.com/2017/09/07/business/equifax-cyberattack.html.

[9] Henry Birge-Lee, Yixin Sun, Anne Edmundson, Jennifer Rexford, and Prateek Mittal. 2018. Bamboozling Certificate Authorities with BGP. In *27th USENIX Security Symposium (USENIX Security 18)*. USENIX Association, Baltimore, MD, 833–849. https://www.usenix.org/conference/usenixsecurity18/presentation/birge-lee

[10] Benjamin Braun. 2016. *Investigating DNS Hijacking Through High Frequency Measurements*. Technical Report. UC San Diego. https://escholarship.org/uc/item/8tm5c7r7.

[11] CAIDA. 2020. Routeviews Prefix to AS mappings Dataset for IPv4 and IPv6. http://www.caida.org/data/routing/routeviews-prefix2as.xml.

[12] CAIDA. 2021. Inferred AS to Organization Mapping Dataset. https://www.caida.org/data/as_organizations.xml.

[13] CAIDA and Ian Foster. 2021. CAIDA-DNS Zone Database (DZDB). https://dzdb.caida.org.

[14] Censys. 2021. Censys Bulk Data Access. https://censys.io/data.

[15] Alberto Cerpa and Jeremy Elson. 2003. Internet Content Adaptation Protocol (ICAP). RFC 3507. https://doi.org/10.17487/RFC3507 https://rfc-editor.org/rfc/rfc3507.txt.

[16] Cloudflare. 2018. BGP leaks and cryptocurrencies. https://blog.cloudflare.com/bgp-leaks-and-crypto-currencies/.

[17] Let's Encrypt Community. 2017. Why no CRL URL in the certificate? https://community.letsencrypt.org/t/why-no-crl-url-in-the- certificate/25686.

[18] David Dagon. 2008. DNS Poisoning: Developments, Attacks and Research Directions. USENIX Security 2008, DNS Panel Talk. https://www.usenix.org/legacy/events/sec08/tech/slides/dagon_slides.pdf.

[19] Matt Dahl. 2019. Widespread DNS Hijacking Activity Targets Multiple Sectors. https://www.crowdstrike.com/blog/widespread-dns- hijacking-activity-targets-multiple-sectors/.

[20] Department of Homeland Security. 2019. Emergency Directive 19-01: Mitigate DNS Infrastructure Tampering. https://cyber.dhs.gov/ed/19-01/.

[21] Digicert. 2021. Domain Control Validation (DCV) Methods. https://docs.digicert.com/manage-certificates/dv-certificate-enrollment/domain-control-validation-dcv-methods/#dns-txt-validation

[22] Digicert. 2022. What's The Difference Between DV, OV & EV SSL Certificates? https://www.digicert.com/difference-between-dv-ov- and-ev-ssl-certificates.

[23] DomainTools. 2022. Iris Investigation Platform - Passive DNS. https://www.domaintools.com/products/iris.

[24] Zakir Durumeric. 2021. Censys Search 2.0. https://support.censys.io/hc/en-us/articles/360060941211-Censys-Search-2-0-Official-Annou ncement.

[25] Digital Element. 2021. NetAcuity IP Geolocation Data. https://www.digitalelement.com/geolocation/.

[26] Let's Encrypt. 2021. Challenge Types - DNS-01 Challenge. https://letsencrypt.org/docs/challenge-types/.

[27] Entrust. 2019. What is a SAN (Subject Alternative Name) and how is it used? https://www.entrust.com/blog/2019/03/what-is-a-san-and- how-is-it-used/.

[28] Gandi. 2022. How to Turn On Transfer Lock for a Domain. https://docs.gandi.net/en/domain_names/transfer_out/transfer_lock.html.

[29] Google. 2017. Broadening HSTS to secure more of the Web. https://security.googleblog.com/2017/09/broadening-hsts-to-secure-more- of-web.html.

[30] Google. 2017. Next steps toward more connection security. https://blog.chromium.org/2017/04/next-steps-toward-more-connection.html.

[31] Google. 2021. A safer default for navigation: HTTPS. https://blog.chromium.org/2021/03/a-safer-default-for-navigation-https.html.

[32] The Guardian. 2017. WikiLeaks hacked as OurMine group answers "hack us" challenge. https://www.theguardian.com/technology/2017/aug/31/wikileaks-hacked -ourmine-group-julian-assange-dns-attack.

[33] Muks Hirani, Sarah Jones, and Ben Read. 2019. Global DNS Hijacking Campaign: DNS Record Manipulation at Scale. https://www.mandiant.com/resources/global-dns-hijacking-campaign-dns-record- manipulation-at-scale.

[34] Rebekah Houser, Shuai Hao, Zhou Li, Daiping Liu, Chase Cotton, and Haining Wang. 2021. A Comprehensive Measurement-based Investigation of DNS Hijacking. In *40th International Symposium on Reliable Distributed Systems (SRDS)* (Virtual Event). IEEE, Chicago, IL, USA, 210–221. https://doi.org/10.1109/SRDS53918.2021.00029

[35] Brian Krebs. 2019. A Deep Dive on the Recent Widespread DNS Hijacking Attacks. https://krebsonsecurity.com/2019/02/a-deep-dive- on-the-recent-widespread-dns-hijacking-attacks.

[36] Brian Krebs. 2020. Does your domain have a Registry Lock? https://krebsonsecurity.com/2020/01/does-your-domain-have-a-registry- lock/.

[37] Ivan Kwiatkowski and Pierre Delcher. 2021. DarkHalo after SolarWinds: the Tomiris connection. https://securelist.com/darkhalo-after -solarwinds-the-tomiris-connection/104311/.

[38] Ben Laurie, Adam Langley, Emilia Kasper, Eran Messeri, and Rob Stradling. 2021. Certificate Transparency Version 2.0. RFC 9162. https://doi.org/10.17487/RFC9162

[39] Baojun Liu, Chaoyi Lu, Haixin Duan, Ying Liu, Zhou Li, Shuang Hao, and Min Yang. 2018. Who Is Answering My Queries: Understanding and Characterizing Interception of the DNS Resolution Path. In *27th USENIX Security Symposium (USENIX Security 18)*. USENIX Association, Baltimore, MD, 1113–1128. https://www.usenix.org/conference/usenixsecurity18/presentation/liu-baojun

[40] Daiping Liu, Shuai Hao, and Haining Wang. 2016. All Your DNS Records Point to Us: Understanding the Security Threats of Dangling DNS Records. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, Vienna, Austria, 1414–1425. https://doi.org/10.1145/2976749.2978387

[41] Keyu Man, Zhiyun Qian, Zhongjie Wang, Xiaofeng Zheng, Youjun Huang, and Haixin Duan. 2020. DNS Cache Poisoning Attack Reloaded: Revolutions with Side Channels. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security* (Virtual Event, USA) *(CCS '20)*. Association for Computing Machinery, New York, NY, USA, 1337–1350. https://doi.org/10.1145/3372297.3417280

[42] Robert McMillan and Dustin Volz. 2021. Suspected Russian Hack Extends Far Beyond SolarWinds Software, Investigators Say. https://www.wsj.com/articles/suspected-russian-hack-extends- far-beyond-solarwinds-software-investigators-say-11611921601.

[43] Warren Mercer and Paul Rascagneres. 2018. DNSpionage Campaign Targets Middle East. https://blog.talosintelligence.com/2018/11/dnspionage-campaign-targets-middle-east.html.

[44] Paul Mockapetris. 1987. Domain Names - Implementation and Specification. RFC 1035. https://rfc-editor.org/rfc/rfc1035.txt.

[45] Elizabeth Montalbano. 2022. Thousands of Malicious npm Packages Threaten Web Apps. https://threatpost.com/malicious-npm-packages -web-apps/178137/.

[46] Audrey Randall, Enze Liu, Ramakrishna Padmanabhan, Gautam Akiwate, Geoffrey M. Voelker, Stefan Savage, and Aaron Schulman. 2021. Home is Where the Hijacking is: Understanding DNS Interception by Residential Routers. In *Proceedings of the 21st ACM Internet Measurement Conference* (Virtual Event) *(IMC '21)*. Association for Computing Machinery, New York, NY, USA, 390–397. https://doi.org/10.1145/3487552.3487817

[47] Eric Rescorla. 2000. HTTP Over TLS. RFC 2818. https://doi.org/10.17487/RFC2818 https://rfc-editor.org/rfc/rfc2818.txt.

[48] Eric Rescorla. 2018. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446. https://doi.org/10.17487/RFC8446 https://www.rfc-editor.org/info/rfc8446.

[49] Rick Lamb. 2022. DNSSEC Deployment Report. http://rick.eng.br/dnssecstat/.

[50] Scott Rose, Matt Larson, Dan Massey, Rob Austein, and Roy Arends. 2005. DNS Security Introduction and Requirements. RFC 4033. https://doi.org/10.17487/RFC4033 https://www.rfc-editor.org/info/rfc4033.

[51] Sectigo. 2022. crt.sh - Certificate Search. https://crt.sh/.

[52] Andreas Sfakianakis. 2020. On Sea Turtle campaign targeting Greek governmental organisations. https://www.linkedin.com/pulse/sea- turtle-campaign-targeting-greek-governmental-andreas-sfakianakis/.

[53] Aftab Siddiqui. 2022. KlaySwap – Another BGP Hijack Targeting Crypto Wallets. https://www.manrs.org/2022/02/klayswap-another -bgp-hijack-targeting-

crypto-wallets/.
[54] Internet Society. 2018. What Happened? The Amazon Route 53 BGP Hijack to Take Over Ethereum Cryptocurrency Wallets. https://www.internetsociety.org/es/blog/2018/04/amazons-route-53-bgp-hijack/.
[55] Sooel Son and Vitaly Shmatikov. 2010. The Hitchhiker's Guide to DNS Cache Poisoning. In *Security and Privacy in Communication Networks*. Springer Berlin Heidelberg, Berlin, Heidelberg, 466–483.
[56] Cisco Talos. 2019. Sea Turtle keeps on swimming, finds new victims, DNS hijacking techniques. https://blog.talosintelligence.com/2019/07/sea-turtle-keeps-on-swimming.html.
[57] The Washington Post. 2013. The New York Times Web site was taken down by DNS hijacking. Here's what that means. https://www.washingtonpost.com/news/the-switch/wp/2013/08/27/the-new-york-times-web-site-was-taken-down-by-dns-hijacking- heres-what-that-means/.
[58] United States of America v Zhang et al. 2017. Case No 13CR3132-H, Indictment (superseding). https://www.justice.gov/opa/press-release/file/1106491/download.
[59] Verisign. 2022. Registry Lock Service. https://www.verisign.com/en_US/channel-resources/domain-registry-products/registry-lock/index.xhtml.
[60] Verizon. 2021. Data Breach Investigations Report 2021. https://enterprise.verizon.com/resources/reports/2021-data-breach- investigations-report.pdf.
[61] VirusTotal. 2021. VirusTotal update_mfa.exe Details. https://www.virustotal.com/gui/file/8900cf88a91fa4fbe871385c8747c7097537f1b5f4a003418d84c01dc383dd75/.
[62] Dustin Volz. 2019. DNC Says Russia Tried to Hack Into its Computer Network Days After 2018 Midterms. https://www.wsj.com/articles/dnc-says-russia-tried-to-hack-into-its-computer-network-days-after -2018-midterms-11547831410.
[63] Lan Wei and John Heidemann. 2020. *Whac-A-Mole: Six Years of DNS Spoofing*. Technical Report. University of Southern California. https://arxiv.org/pdf/2011.12978.pdf.
[64] Florian Weimer. 2005. Passive DNS Replication. https://www.first.org/conference/2005/papers/florian-weimer-paper-1.pdf.

```
<div id="errorMessageDiv" class="errorMessage">
  Для продолжения работы с сервисом электронной почты
    необходимо установить обновление безопасности:
  <br>
<a href="update-mfa.exe">Скачать обновление</a></div>
```

Figure 6: Error message added to the counterfeit `mail.mfa.gov.kg` site to trick users into installing malware. Translation courtesy Google Translate: "To continue using the email service, you must install the security update: Download Update".

Data Sets	Measurement Period	Access
Censys CUIDS [14]	Jan 2017 - Mar 2021	Research Access
CAIDA pfx2as [11]	Jan 2017 - Mar 2021	Research Access
NetAcuity Geolocation [25]	Jan 2017 - Mar 2021	Research Access
Services	**Measurement Period**	**Access**
DomainTools pDNS [23]	-	Research Access
CAIDA DZDB [13]	-	Open Access
crt.sh Certificate Search [51]	-	Open Access

Table 6: List of data sets and services used in this study. The measurement period notes the time frame from which data was used. For services, only data for shortlisted domains around specific times is queried. Access indicates what type of access we had to the data set or service.

A EVOLUTION OF KYRGYZSTAN HIJACKS

Starting late 2020, Censys started collecting additional service and device context [24] including HTTP Responses. This additional context provides further visibility into the nature and evolution of these attacks. For example, during our study we observed that visitors to `mail.mfa.gov.kg` (a Zimbra-based mail login page) were redirected to a site in Russia that also provided a (valid) counterfeit certificate for the site — consistent with the well-established strategy of harvesting credentials typed into these web pages. Indeed, using Censys' HTTP response data we were then able to verify that while the page mimicked the Zimbra login page's look and feel, it differed from the standard Zimbra code. Moreover, in May 2021, we observed the domain redirected to *a new* IP: 178.20.46.22 (again originated by AS 48282 in Russia). This server also mimicked the Zimbra mail login page, but included *additional JavaScript code* to render an error message (shown in Figure 6) intended to socially engineer users into installing the "security update" software: *update-mfa.exe*. We identified this executable on VirusTotal, uploaded shortly after the redirection is initiated [61]. This software, written in Go, has since been identified as the downloader for the *Tomiris* implant, itself loosely associated with the SolarWinds attack [37]. We surmise that the attacker found that credentials interception alone was insufficient for their needs (*e.g.*, perhaps due to the use of multi-factor authentication for some users).

B SUPPLEMENTARY INFORMATION

Table 6 lists the data sets and services used in this study. The measurement period notes the time frame from which data was used. For services, data is only queried for shortlisted domains around specific times of interest *i.e.*, around the weeks of suspicious transient deployments. It also lists if the type of access we had to the data set or service. While CAIDA-DZDB (TLD zone files) and crt.sh (CT Logs) are open access (though rate-limited), the other data sets and services require applying for research access.

Table 7 lists the 41 hijacked domains including the broad sector level categorization of the organization behind the domains. Similarly, Table 8 lists the 24 targeted domains along with the organization and the broad sector level categorization.

Table 9 lists the suspiciously obtained certificates along with their revocation status (if available) for the 40 hijacked domains that used TLS certificates.

CC	Domain	Description	Sector
AE	adpolice.gov.ae	Abu Dhabi Police, UAE	Law Enforcement
AE	apc.gov.ae	Police College Website, UAE	Law Enforcement
AE	mgov.ae	Telecommunications Regulatory Authority, UAE	Government Organization
AE	mofa.gov.ae	Ministry of Foreign Affairs, UAE	Government Ministry
AL	asp.gov.al	Albanian State Police, Albania	Law Enforcement
AL	e-albania.al	E-Government Portal, Albania	Government Internet Services
AL	shish.gov.al	State Intelligence Service, Albania	Intelligence Services
CY	cyta.com.cy	Telecommunications Provider, Cyprus	Infrastructure Provider
CY	defa.com.cy	Natural Gas Public Company, Cyprus	Energy Company
CY	govcloud.gov.cy	Government Internet Services, Cyprus	Government Internet Services
CY	owa.gov.cy	Government Internet Services, Cyprus	Government Internet Services
CY	sslvpn.gov.cy	Government Internet Services, Cyprus	Government Internet Services
CY	webmail.gov.cy	Government Internet Services, Cyprus	Government Internet Services
EG	mfa.gov.eg	Ministry of Foreign Affairs, Egypt	Government Ministry
EG	mod.gov.eg	Ministry of Defense, Egypt	Government Ministry
EG	nmi.gov.eg	National Institute for Governance, Egypt	Government Organization
EG	petroleum.gov.eg	Petroleum and Mineral Wealth Ministry, Egypt	Government Ministry
GR	kyvernisi.gr	Government Internet Services, Greece	Government Internet Services
GR	mfa.gr	Ministry of Foreign Affairs, Greece	Government Ministry
IQ	mofa.gov.iq	Ministry of Foreign Affairs, Iraq	Government Ministry
IQ	inc-vrdl.iq	E-Government Portal, Iraq	Government Internet Services
JO	gid.gov.jo	General Intelligence Directorate, Jordan	Intelligence Services
JO	psd.gov.jo	Public Security Directorate, Jordan	Intelligence Services
KG	fiu.gov.kg	Financial Intelligence Service, Kyrgyzstan	Government Ministry
KG	invest.gov.kg	Investment Portal, Kyrgyzstan	Government Ministry
KG	mfa.gov.kg	Ministry of Foreign Affairs, Kyrgyzstan	Government Ministry
KG	infocom.kg	Internet Services	Infrastructure Provider
KW	csb.gov.kw	Central Statistical Bureau, Kuwait	Government Ministry
KW	dgca.gov.kw	Directorate General of Civil Aviation, Kuwait	Civil Aviation
KW	kotc.com.kw	Kuwait Oil Tanker Company	Energy Company
KW	moh.gov.kw	Ministry of Health, Kuwait	Government Ministry
LB	finance.gov.lb	Ministry of Finance, Lebanon	Government Ministry
LB	mea.com.lb	Middle East Airlines, Lebanon	Civil Aviation
LB	medgulf.com.lb	Insurance Company, Lebanon	Insurance
LY	embassy.ly	Libyan Embassies	Government Organization
LY	foreign.gov.ly	Ministry of Foreign Affairs, Libya	Government Ministry
LY	noc.ly	National Oil Corporation, Libya	Energy Company
NL	ocom.com	Internet Services	Infrastructure Provider
SE	netnod.se	Internet Services	Infrastructure Provider
SY	syriatel.sy	Telecommunications Provider, Syria	Infrastructure Provider
US	pch.net	Internet Services	Infrastructure Provider

Table 7: Description of 41 domains identified as hijacked including the broad sector level categorization.

CC	Domain	Description	Sector
AE	epg.gov.ae	Emirates Post, UAE	Postal Service
AE	milmail.ae	Armed Forces Mail, UAE	Law Enforcement
AE	mocaf.gov.ae	Ministry of Cabinet Affairs, UAE	Government Ministry
AE	moi.gov.ae	Ministry of Interior, UAE	Government Ministry
CH	parlament.ch	Parliament, Switzerland	Government Organization
GH	nita.gov.gh	National Information Technology Agency, Ghana	Government Organization
KZ	zerde.gov.kz	National Infocommunication Holdings, Kazakhstan	Government Organization
LB	pcm.gov.lb	Presidency of the Council of Ministers, Lebanon	Government Ministry
LT	stat.gov.lt	Statistics Lithuania	Government Ministry
LV	iem.gov.lv	Ministry of the Interior, Latvia	Government Ministry
LV	zva.gov.lv	State Agency of Medicines, Latvia	Government Organization
MA	justice.gov.ma	Ministry of Justice, Morocco	Government Ministry
MA	mem.gov.ma	Ministry of Sustainable Development, Morocco	Government Ministry
MM	mofa.gov.mm	Ministry of Foreign Affairs, Myanmar	Government Ministry
PL	knf.gov.pl	Polish Financial Supervision Authority	Government Ministry
SA	cmail.sa	Al-Elm Information Security	IT Firm
TM	turkmenpost.gov.tm	Turkmen Post	Postal Service
US	batesvillearkansas.gov	City of Batesville, AR	Local Government
US	manchesternh.gov	City of Manchester, NH	Local Government
VN	ais.gov.vn	Authority of Information Security, Vietnam	Government Organization
VN	cpt.gov.vn	Central Post Office, Vietnam	Postal Service
VN	mofa.gov.vn	Ministry of Foreign Affairs, Vietnam	Government Ministry
VN	most.gov.vn	Ministry of Science and Technology, Vietnam	Government Ministry
VN	vass.gov.vn	Vietnam Academy of Social Sciences	Government Organization

Table 8: Description of 24 domains identified as targeted including the broad sector level categorization.

CC	Domain	Target	crt.sh ID	Issuer CA	CRL
AE	adpolice.gov.ae	advpn	835334320	Let's Encrypt	—
AE	apc.gov.ae	mail	820893483	Let's Encrypt	—
AE	mgov.ae	mail*	804429558	Let's Encrypt	—
AE	mofa.gov.ae	webmail	495595690	Comodo	✖
AL	asp.gov.al	mail*	929142682	Comodo	✔
AL	e-albania.al	owa	296537802	Let's Encrypt	—
AL	shish.gov.al	mail	912593168	Let's Encrypt	—
CY	cyta.com.cy	mbox	1150009364	Comodo*	✔
CY	defa.com.cy	mail	1225501249	Comodo*	✖
CY	govcloud.gov.cy	personal*	1021403642	Comodo	✖
CY	owa.gov.cy	.	1056463948	Comodo	✖
CY	sslvpn.gov.cy	.	1088915811	Comodo	✖
CY	webmail.gov.cy	.	1039430428	Comodo	✖
EG	mfa.gov.eg	mail	946136592	Let's Encrypt	—
EG	mod.gov.eg	mail*	970178538	Let's Encrypt	—
EG	nmi.gov.eg	mail*	961982738	Comodo	✖
EG	petroleum.gov.eg	mail	962230186	Let's Encrypt	—
GR	kyvernisi.gr	mail	1394170951	Let's Encrypt	—
GR	mfa.gr	pop3	1382284606	Let's Encrypt	—
IQ	mofa.gov.iq	mail	775703946	Let's Encrypt	—
IQ	inc-vrdl.iq	.	961752433	Let's Encrypt	—
JO	gid.gov.jo	.	1024142638	Let's Encrypt	—
KG	fiu.gov.kg	mail	3848797679	Let's Encrypt	—
KG	invest.gov.kg	mail	3842234495	Let's Encrypt	—
KG	mfa.gov.kg	mail	3810274168	Let's Encrypt	—
KG	infocom.kg	mail	3913246526	Let's Encrypt	—
KW	csb.gov.kw	mail	2288836441	Let's Encrypt	—
KW	dgca.gov.kw	mail	291715835	Let's Encrypt	—
KW	kotc.com.kw	mail2010*	1485763752	Let's Encrypt	—
KW	moh.gov.kw	webmail	1394227599	Let's Encrypt	—
LB	finance.gov.lb	webmail	922787324	Let's Encrypt	—
LB	mea.com.lb	memail	923463031	Let's Encrypt	—
LB	medgulf.com.lb	mail	983855608	Let's Encrypt	—
LB	pcm.gov.lb	mail1	983220130	Let's Encrypt	—
LY	embassy.ly	.	—	—	—
LY	foreign.gov.ly	.	893184607	Let's Encrypt	—
LY	noc.ly	mail	885156392	Let's Encrypt	—
NL	ocom.com	connect	314340862	Comodo	✖
SE	netnod.se	dnsnodeapi	1071765455	Comodo	✔
SY	syriatel.sy	mail	1349974775	Let's Encrypt	—
US	pch.net	keriomail	1075482666	Comodo	✔

Table 9: List of suspiciously obtained certificates for 40 hijacked domains (`embassy.ly` did not use TLS certificates.) Let's Encrypt is the Issuer CA for 28 while Comodo is the Issuer CA for 12. Both of the CAs provided certificates for free. Only 4 certificates were revoked. Let's Encrypt does not provide a CRL for the leaf certificates and instead relies on Online Certificate Status Protocol (OCSP). As a result, we cannot determine retroactively if any of the 28 certificates issued by Let's Encrypt were revoked.