IAB Programme
Internet-Draft
Intended status: Informational
Expires: July 6, 2022

D. Lazanski
Last Press Label

January 6, 2022

Security Considerations for Protocol Designers
draft-lazanski-protocol-sec-design-model-t-04.txt

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on July 6, 2022.

Copyright Notice

Abstract

This document is a non-exhaustive set of considerations for protocol designers and implementers with regards to attack defence. This document follows on from the way forward outlined in draft-lazanski-users-threat-model-t-04. These considerations both supplement and support the work on threat models. They can be used as an aid to analyse protocol design choice and in turn to help combat threats and defend users of these protocols and systems against malicious attacks.

First, we list well-known classes of attacks that pose threats, with relevant case studies and descriptions. Next, we give a list of defence measures against these attacks to be considered when designing and deploying protocols. Naturally, deployments of protocols vary greatly between use cases; therefore, some attacks and defensive measures outlined may require more consideration than others, dependent on use case.

This RFC can be used by protocol designers to write the Security Considerations section in an RFC. The impact on attack defence of a protocol should be considered in multiple use cases across the multiple layers of the internet. Defence against malicious attacks can be improved and it can be weakened by design features of protocols. Designers should acknowledge the role of protocols in attack prevention, detection and mitigation; this document aims to be a useful guide in doing so.

Table of Contents

1. Introduction

This draft aims to give a non-exhaustive set of attack defence considerations for protocol designers and implementers to consider when designing and deploying protocols. These considerations are focused on informing the design of protocols so that protocols may better defend users and systems from malicious attacks.[1] This is essential information and should be considered for protocol development. No protocols should be finalised without security guidance just as no protocols should be designed without privacy considerations. This document is a useful and necessary reference and it is the intention of the authors that the IETF makes full use of all the security expertise in its community through the updating of this document.

For protocol designers, it is important that a protocol's impact on different attack defence cases across the layers of the internet should be considered. Defence against malicious attacks can be either improved or weakened by the design features of protocols.

Designers should acknowledge that including attack prevention, detection and mitigation is essential in protocol development.

In Section 2, we list some of the many attacks that are a malicious presence on the internet today, with their methodologies, notable case studies and attack outcomes. This is not, and can never be, an exhaustive list; threats have been chosen based on their frequency and regularity, likelihood of occurring, and impact on victims. In Section 3, we describe some real world impacts following up on some of the attacks listed in Section 2.

In Section 4, we document some known popular current defensive practice, giving a list of defence measures that can and are widely used against attacks from Section 2. These current practices should be considered when designing and deploying protocols to avoid obsoleting them unwittingly. Other possible defensive measures for protocol designs are included where relevant.

Section 5 outlines the motivations and use of this draft, and Section 6 suggests the methodology by which considerations outlined in this draft may be consistently thought through in protocol design.

2. Attacks

In this section we outline some attack types that are well-known in industry and in public. For each attack type, we give examples of how this attack is carried out, case studies of notable attacks where appropriate, and the outcomes that attackers are trying to achieve. Some considerations for protocol designers in relation to each specific attack are also listed.

Throughout this draft the aim to use common industry references, taxonomies and terminology for types of attack to avoid confusion. Considerations for protocol designers and deployments in each section are summarised in a table in Section 2.11 for ease of reference.

2.1. Endpoint Compromise

According to IDC 70% of successful cybersecurity breaches originate at the endpoint as their initial point of contact and penetration. Attack Description: Endpoint compromise is when a malicious an unauthorised attacker has control of an endpoint beyond their access and privilege level. Endpoint compromises not yet been mentioned in RFC 3552. However, they may be achieved through many attack vectors, such as: stealing legitimate credentials that give access to the endpoint, malware infection, a phishing attack, and insider threats.

Attackers have multiple motivations to compromise an endpoint, some of which we list here and include: to exfiltrate personal or intellectual proper data on the endpoint, to perform reconnaissance for other malware to deploy, to move laterally around a network, to harvest legitimate credentials, for financial gain, or for reputational or political reasons. See draft-lazanski-users-threat-model-t

When an endpoint is compromised, it is common practice to utilize a communication channel (either a new one is opened or an existing one is leveraged) to exfiltrate data, communicate to the command centre, explore the network or propagate to other endpoints. Such a communication channel would typically attempt to "look like" a routine connection to a server or a peer. Furthermore, malware often disables any protection, if any protection exits, on the endpoint as part of its initial infection process allowing this to happen quickly.

Case Study: Silex Malware

In June 2019 a strain of malware was found that wipes the firmware of an IoT device. It does this by using known credentials for logging into IoT devices and completely wipes the system and removes the network configuration. It impacted thousands of devices by rendering them useless. [2]

Protocol Design Considerations: A protocol design consideration against this attack would therefore preserve prevention mechanisms and allow for the detection of the abnormality of connections on host systems. This allows for network-based defence in depth. Abnormalities might include unusual traffic flow to a server, attempting to contact many IP addresses (scanning), or beaconing behavior patterns, which is when it 'calls home' at regular intervals.

This is just one example of potential endpoint compromise situations and subsequent mitigations which can be considered and included in protocol development. More detailed consideration of endpoints, especially with respect to endpoint-only security solutions can be found in Capabilities and Limitations of an Endpoint-only Security Solution draft-taddei-smart-cless introduction-02 and a related taxonomy, Endpoint Taxonomy for CLESS draft-mcfadden-smart-endpoint-taxonomy-for-cless-01 which is a taxonomy of specific endpoint equipment, devices and applications. These drafts both highlight important points. In the draft-taddei-smart-cless-introduction-02 researchers found that 5% of incidents were only detected by network based systems while the rest was detected by endpoint security. This

is why the reliance on network-based protocols shouldn't be the only focus in protocol design.

2.2. Network Compromise

Attack Description: Network compromise is where a malicious and unauthorised attacker has presence on or control of part of a network beyond their privilege level. An attacker may compromise a network to achieve any one of many objectives, such as: to obscure endpoint compromise, to move laterally around a network undetected, to perform reconnaissance, to gain information or data and to escalate privilege.

Case Study: NotPetya

The NotPetya virus initiated a series of global, sustained attacks in 2017 which originated in the Ukraine, but spread rapidly. A variant of Petya ransomware virus, NotPetya targets Windows based systems in order to infect the master boot loader which in turn encrypts the hard drive's file system. However, unlike Petya, NotPetya spreads on its own through network compromise and encrypts everything. Also, it looks like Petya ransomware, but is in fact not ransomware (hence the name NotPetya). Though the attack was global, the global shipping and logistics company Maersk lost their entire IT system which impacted their business. The estimated loss to their business was around 300 million Euros.[3]

Protocol Design Considerations:

If there is an unauthorised attacker accessing a passive inspection point in the network, a protocol design consideration would be to apply cryptography for confidentiality protection.

A protocol design consideration for an attacker modifying contents of data packets on the network would be to apply cryptography for integrity protection.

Finally, if an attacker engages in re-routing, re-ordering, replaying, delaying or dropping data on the network, which is arguably less well-handled by existing Internet transport protocols, then protocol design considerations would include development of strong sender authentication, integrity checks over whole sessions not just individual packets, replay detection, and out-of-order packet detection.

2.3. Denial of Service (DoS) and Distributed Denial of Service (DDos)

Attack Description: Denial of Service (DoS) attacks intend to prevent any service and service delivery. Attackers usually select a high-profile, online web target that they intend to make unavailable in the short-term. Distributed DoS attacks utilise multiple compromised endpoints to distribute the DoS attack, removing the possibility of blocking the attack by removing the single device launching the attack.

DDoS attacks can occur:

1) at the network layer, known as a Layer 2 DDoS attack, launched via malformed packets, flooding or spoofing.

2) at the application layer, known as a Layer 7 DDoS attack, launched via Ping of Death, HTTP floods, XDoS.

3) a combination of both network and application services, resulting in amplification and reflection attacks.

Common DDoS Attacks include:

1) HTTP POST attack in which an attack floods an HTTP POST or GET request by exploiting an open connection and sending data to connected web servers, typically over a period of time. This takes place at Layer 7 and is successful because it is an asymmetric attack which leaves the connection open for a long duration. An update would be required to fix this issue.

2) ICMP flood or ping flood is when an attacker takes down a computer by sending a large number of request packets. This is accomplished by knowing the destination IP address and sending the requests.

A successful DDoS attack, where the service is inaccessible for a period of time, achieves the attacker objective include degradation of service. In some cases, this may be used by the attacker as an opportunity for extortion.

Case study: Dyn attack, Mirai

The Mirai botnet was first identified in 2016. The Mirai botnet as well as variants target and infect Internet of Things devices. Those infected devices scan the Internet for IP addresses of other Internet of Things devices, thus creating a multiplication of IoT devices which are infected. Though the bot still exists in various forms, the most serious attack took place on 21 October 2016 when

the Domain Name System (DNS) provider Dyn was attacked by DDoS using a coordinated system of infected IoT devices. Much of the Internet was unreachable after three attacks occurred during the same day. The decentralised nature of the Internet helped to mitigate the severity of attack and the attack was eventually resolved on that day. However, the sheer size and scale of the attack is still viewed as one of the biggest attacks on the Internet to take place.. [4] Changes in the threat landscape, including risk of consolidation and IoT changes could upend these mitigations in future and promote over-reliance on one single security solution rather than a decentralized, resilient network.

Protocol Design Considerations: Some people take the attitude of "DDoS attack? Welcome to the internet" and this is the approach of RFC 3552 as well. However, the use of the Internet and data that travels over it has increased exponentially. Protocol designers should be aware of potential issues that help DDoS attacks, such as: CoAP flooding, protocols that permit mass unscheduled deliveries, provision of the ability for an attacker to mask e.g. IP addresses, provision of the ability to amplify, a lack of sender authentication, a long time open request and an inability to filter at scale.[5] Assessment of protocol for abuse and DoS amplification should be a part of the security assessment during the design iteration process.

2.4. Phishing

Attack Description: A phishing attack is where an unsolicited message with malicious content is received. Malicious content could be either in the message itself (email, messages), or directing the user to a malicious domain. Varieties of phishing exist, based on difference social engineering approaches, including: spear phishing, clone phishing and whaling. Phishing is cited as the initial attack vector for 91% of reported malicious attacks.[6]

For a phishing attack to succeed, the user has to be unwittingly duped into an action, where they can't be assumed to have the knowledge to check their action. For example, users can be confused by domain names that render almost identically but are different at a binary level, such as internationalised domain names. Also users may see that a sender of an email to them is someone they know, but not realise that the email address is different.

Case Study: Ukrainian Power Grid Attack

The cyber attack on the Ukrainian Power Grid took place on December 23, 2015. It was the first known cyber attack on a power grid. Around 250,000 individual customers were impacted. DDoS telephone

attacks also prevented people from calling help centres to report the lack of electricity. All of this began with a spear-phishing campaign initiated 9 months before that was eventually successful. [7]

Protocol Design Considerations: design protocols with strong authentications and identification/proof of ownership of domains required. Provide users with means to assist checking their actions are safe (or automate the means that can check a user's actions).

Network infrastructure should be able to detect whether data has strong authentication and policies can be specified for handling unauthenticated data (e.g. SPF, DMARC). One defensive measure employed by domain owners to check for unauthorised usage of identical or similar domain names is to use Certificate Transparency logs [8] with automated notifications to the domain owner where domains with 'close' domain names log a certificate, indicating a malicious spoofed domain and therefore access is denied. [9]

2.5. Malware Infection

Attack Description: Malware is one attack vector used to achieve network or endpoint compromise. As Lockheed Martin's Killchain describes,[10] there are many interactions between malware and protocols which allows for infection and attacks. In the case of Lockheed Martin there are seven distinct stages each with protocols associated with them. Malware comes in many different strains and flavours: exploit kits, ransomware, viruses, trojans, worms, rootkits and more. The attack outcomes are incredibly varied too - attackers might want to recruit bots, deploy that leaves behind physical damage, steal IPR material for corporate espionage, exfiltrate of credentials to gain accesses elsewhere in the system, perform reconnaissance for a later attack, or make some financial gains.

Case Study: WannaCry

In 2017, a ransom attack was launched by using a cryptoworm targeting computers running the Microsoft operating system. The attack encrypted person and computer data and then asked for a ransom in order to unencrypt the data. The attack was eventually stopped by a 'kill switch' that was discovered. DNS monitoring detected the infection, but not without infecting 200,000 computers first.[11]

Protocol Design Considerations: As malware is often carried to an endpoint by an Internet protocol, there are considerations for protocol designers. Moreover, once it's arrived at its destination, malware needs to use protocols for discovery of peers, for C2, for exfil. Therefore, such connections and the features from those protocols can be used to detect, track and mitigate outbreaks in real-time. For example, SMTP headers can detect malware spreading through e-mail, and other protocol connections can show the lateral movement of malware through a network.

TODO: Details for detection for protocol design.

2.6. Insider Threat

Attack Description: This attack involves social manipulation of an authorised person so that they knowingly attempt malicious actions, using their authorised privileges, credentials and accesses to achieve nefarious attacker objectives. This is different social engineering to phishing (Section 2.4), where the user is unwitting to their actions.

The insider themselves might be the sole attacker, for various reasons - ranging from a desire to gain notoriety or to inflict deliberate harm on their employer. If the insider is manipulated by another attacker, their role is likely to provide information only accessible by an outsider, to enable further attacks. Eventual attacker outcomes are to gain access to the network or endpoints, potentially for insider trading, fraudulent transactions or IPR theft.

Case Study: Anthem

A contractor working for a consultancy employed by Anthem stole 18,500 individual files with personal details and used them for personal gain.[12]

Protocol Design Consideration: There is therefore a need for authorization to be a design consideration for protocols, and also provisioning the ability to create and manage logs, and to create audit trails for document access.

TODO: Fill in authorization process and abnormality detection for protocol design.

2.7. Hijacking Traffic

Attack Description: Border Gateway Protocol hijacking, or BGP hijacking is when a group of IP addresses are taken over maliciously

by routing table manipulation and corruption. BGP hijacking is fairly common and is a frequent attack used against cryptocurrencies because hosting centralization makes it particularly vulnerable to attacks. As recently as June 2019 a large amount of European Internet traffic was re-routed through China because of a BGP route leak. However, instead of China telecom ignoring the leak it hijacked the routes as their own. [13]

Case Study: In 2018 Amazon Web Services DNS offering called Amazon's Route 53 was hijacked by using BGP table updates which directed the traffic to a malicious server at an IXP in Chicago. The attack lasted two hours and resulted in stolen Ethereum cryptocurrency from myetherwallet.com [14]

Protocol Design Considerations: Protocol design considerations should include authentication and handshake management when sending and accepting traffic. This will be an ongoing and iterative process so the protocol design must take into consideration the management of this repetitive process. Protocol design should also take into account how to prevent DNS cache poisoning and route table manipulation and communicate that such a process occurred.

2.8. Web-based Attacks

Attack Description: Web-based attacks are those that use web systems and services as the main surface for compromising the victim [15] including browser exploitations, like the Firefox zero day exploit found in the new version of Mozilla's browser in January 2020 [16] and injections, drive-by downloads, cross-site request forgery, water-holing, and more. Web-based attacks are on the rise and Web application attacks also continue in the form of malicious web applications, SQL injections and cross-site scripting. [17]

Case Study: Chrome

As recently as February 2020 a security vulnerability in older versions of the Chrome browser allowed for the exploitation of user's computers in a zero day attack scenario. Though the vulnerability has been fixed through updates, a number of attacks have taken place. Number unknown to date. [18]

Protocol Design Considerations: Many mitigations to these attacks rely on endpoint security, such as patching. This may possibly explain the rise in this attack trend. One simple way to mitigate this attack vector is to make patching updates as easy and straightforward as possible. This includes clear communication for when updates are needed and how the user can safely and securely patch and update. Protocol designers should be aware of other

mitigations, such as web-traffic filtering and web-traffic encryption, in order to take them into consideration.

2.9. Malware Free Attacks

Attack Description: Malware-free attacks accounted for 51% of all global cyberattack types, according to Crowdstrike's 2020 Global Threat Report [19] for the first time since starting the report. A malware free attack is one that does not employ or write a malicious file or fragment a computer disk. Instead, memory executed code or stolen credentials, running legitimate tools or executing code from memory are all possible types of attacks. Malware-free attacks are more difficult to detect unless actively looking for cyberattacks in systems.

Case Study: TBD TODO

Protocol Considerations: Building in resilience to malware-free cyber attacks. Allow for search and notification of potential cybersecurity issues as a pre-emptive measure.

2.10. Table of Attacks TODO

This section will be a table of attacks, case studies and the relating protocol design considerations. It will updated once all of the case studies have been added.

3. Real World Impacts

The following section focuses on the impacts and outcomes that happen as a result of cyber attacks. This section is by no means comprehensive, but will expand as examples are added through contributions and collaborations with those who are interested.

3.1. Remote Data Alteration

Attack Description: Alteration of data on a remote system, e.g. a Industrial Control System, to achieve an effect that would, for example, change the delivery of products in a supply chain or change the characteristics of a product during production may cause harm to people using that product intended for one use, but designed to malfunction. RDP allows cyber attacks to access the Internet-facing parts of an ICS from where they may able to move to the operational environment.

Case Study: Industrial Control Systems or TBD [20] TODO

Protocol design considerations: Industrial Control Systems are expensive and are often patched in slower-time, and a defence-in-depth approach is advocated; endpoint security alone is not enough. Additional considerations include the ability to real-time monitor easily, and to note that internet protocols are used for high-threat systems.

3.2. Data Exfiltration

Attack Description: Data exfiltration is a frequent outcome of compromise, where data is taken from a system by an unauthorized user. This information leakage includes customer data (often in high-profile breaches) or theft of IPR material from enterprises.

Exfiltration of data can be:

1) High-volume, where the attacker expects to be detected or wants to operate quickly.

2) Low-and-slow, where data is siphoned off at a low level for a long period of time, in the hope of avoiding detection.

Case study: Equifax

In March 2017, attackers searched the web looking for vulnerabilities that were known, but had not been fixed. Making patches easier to download would have easily solved this issues. These attackers targeted the dispute resolution portal at a credit ratings company called Equifax in the US. The hackers used a vulnerability in Apache Struts which allowed access and exfiltration of personal information on the portal. [21]

Protocol Design Considerations: Endpoints can tag/watermark content so that leaked data can be identified and possibly stopped at a gateway, or at least traced back to the user that leaked the material. For this, protocol data could include a protective marking field that is visible to a firewall, even if the content is encrypted.

Another issue to consider is the detection of data exfiltrated through covert channels. Protocols should be designed with this abuse in mind, with designers minimising existence and size of covert channels.

3.3. Identity Theft

Attack Description: Fraud committed from the theft of personal identifiable information - such as bank details, home address and

date of birth - strengthened by the massive digitisation and centralisation of people's personal data. Credential stealing and credential stuffing are two of many ways to obtain personal data.

Case Study: JP Morgan Chase

In 2014 JP Morgan Chase had over 83 million accounts compromised and hackers made over $100 million through fraud and identity theft. To date it is one of the largest data breaches in history.[22]

Protocol Design Considerations: Provision and protect a way that allows breaches of personal data to be detected in real-time and stopped.

4. Defensive Measures

Defensive measures broadly fall into three classes:

1. prevention of attacks - stopping most attacks from achieving the attacker's objectives, i.e. from taking hold on a system, network or endpoint.

2. detection of attacks - how attacks are detected quickly, efficiently, with a high-degree of confidence and accuracy.

3. mitigation of attacks - once an attack happens, the capability to stop the damage done by the attack, e.g. preventing the spread of the compromise within an organisation, limiting the data exfiltrated or stopping the attack from being replicated globally on unaffected systems.

All defences listed in this section relate to one or more attacks listed in Section 2.

For each type of defensive measure, we categorise the method as prevention, detection, mitigation or a combination of these; we link to the type of attack in Section 2 that is prevented; and we describe how the defence work.

Considerations for protocol designers (in relation to each defensive measure) are also listed throughout, and summarised in Table 4.16 to provide an easier reference.

4.1. Response to Attacks

Defence Type: Mitigation

A system can be designed to ensure availability under attack, e.g. by segregating classes of devices on a network, or considering system architecture. Components that are under attack have a channel for reporting that attack that is distinct from the channel used for launching the attack.

Case Study: TBD TODO

Protocol Design Considerations: Design protocols that allow such segregation in architecture in a simple and scalable way. Design protocols for reporting of attacks that use channels that are less susceptible to attacks.

4.2. Recovery from Attacks

Defence Type: Prevention

If organisations and individuals assume that a security breach will happen then defences will be optimised in or to allow for a quick response and minimal impact. This is an important point that is missing from the current version of RFC 3552 because the scale and size of attacks has changed over the years since it was published as noted in draft-mcfadden-rfc3552-research-methodology.

For example, encrypt data when stored so that if it is stolen, the attacker can't decrypt it. For another example, if data is backed up regularly and stored offline, then the threat held by ransomware is minimised.

Case Study: TBD TODO

Protocol Design Considerations: Design protocols that can deliver encrypted payloads to capable endpoints. Practice strong separation of the keys used to encrypt data at rest from the data, with high levels of protection applied to the keys. Design protocols that allow for regular, automatic backup of data minimising the amount of user interaction required.

4.3. Reporting of Attacks

Defence Type: Detection

Logging is an important feature. Multiple logs allow cross referencing and establishing truth data, so it is important to provide logging in multiple places, to detect false reporting by compromised endpoints or networks. Logging allows strengthened

forensics, which reduces the risk of similar attacks in future. Forensic analysis of logs makes it easier to detect and locate attacks, so can be a deterrent to attackers. For this same reason, malicious manipulation of logs to prevent detection is an attractive attacker objective.

Reliable logging helps to find the source of an attack, its spread and what devices and networks have been compromised. Unreliable logging can slow attempts to mitigate it.

Case Study: TBD TODO

Protocol Design Considerations: How a protocol might help/hinder the ability to troubleshoot or have separate logging in multiple places (for truth data), allow for reliable logging across points in a network.

4.4. Sinkholing

Defence Type: Mitigation

Mitigation against where a DDoS attack has already been launched, to prevent a successful attack outcome (i.e. a denial of service).

Case Study: Nitol Botnet

Through counterfeit Microsoft products, Nitol malware infected over 4000 Windows computers primarily in China. The botnet originated from the domain 3322.org which was a dynamic DNS service. The subdomains of 3322.org from which the malware originated, could redirect traffic to additional sites also infected with malware. The malware propagating itself through Internet traffic. [23] At least 70,000 subdomains were infected. Microsoft attempted to disrupt the supply chain of the infected Microsoft products and block the subdomains. [24]

Protocol Design Considerations: Ability to determine whether a connection is likely malicious or not, and filter out malicious connections at speed. Ability to reliably determine the source of a connection and verify that it is accurate and from an address authorised to make the connection. Ongoing monitoring and reporting is necessary.

4.5. Firewalls/Middleboxes/Gateways

Defence Type: Prevention and detection

Firewalls, middleboxes and gateways can be used to inspect traffic and make decisions whether to allow, block or modify data content. These decisions can be based on simple IP address/port/protocol rules, or on deep packet inspection of individual data packets, or use artificial intelligence/machine learning techniques to make more complex decisions based on analysis of traffic over a period of time.

Case Study: TBD TODO

Protocol Design Considerations: Protocols should allow for selective and/or minimally intrusive analysis, balanced against the need to protect the privacy of the user content and any personally identifiable information. Minimally intrusive analysis could include the ability to block traffic that it associated with known insecure protocols or ports, known malicious activity or known malicious users. A protocol that makes all traffic look essentially indistinguishable forces the firewall into making an "all-or-nothing" decision, which would be ineffective for defending against attacks if it is still to allow some communications. Allowing for communication would be detrimental for privacy as well.

A firewall should not be able to undetectably modify traffic; where it is necessary to modify traffic to prevent a threat, the modification should be apparent to the receiver.

4.6. Intrusion Prevention System (IPS) and Intrusion Detection System (IDS)

Defence Type: prevention and detection

These systems provide the abilities to prevent and detect malware infection, vulnerability exploits, and lateral movement on compromised networks and endpoints. Signatures for bad content - IPS/IDSs don't work on traffic if attacks have legitimate content but bad intent, e.g. DDoS. However, the IPS/IDS may detect the malware that compromised the endpoint to launch the DDoS attack.

IDSs are passive systems that scan traffic and report to the traffic owner on threats; IPSs are inline, taking an active role and making automated decisions and applying these actions to traffic.

Case Study: Vishing and Covid-19

IDS and IPS are employed in companies and organisations especially in office and controlled physical environments. However, because of Covid-19 people are working from home or remotely. IDS is used in offices; now people working remotely or working from home don't have

that protection. This has led to an upsurge in vishing. Vishing is when an attacker builds a profile and convinces the victim to download and use remote assistance software which becomes the way the attacker penetrates an organisation. In a remote environment without IDS or IPS this could happen without adequate defence. [25]

Protocol Design Considerations: For IDS, protocols should allow for logging of data relating to the connection, which may include any or all of IP addresses, ports, protocols and payloads, balanced with the requirement to protect privacy of user data. For IPS, protocols should allow for signature/statistical anomaly detection, the ability to selectively drop traffic and respond at efficient scale and speed.

4.7. Upstream Filtering

Defence Type: Mitigation

Content is filtered through a "scrubbing" centre to forward only good traffic. This is provided as a service by e.g. Cloudflare, Akamai.

Case Study: Akamai scrubbing

In August 2019, Akamai opened a new scrubbing centre in Melbourne to compliment its other Asian regional centres in Sydney, Tokyo, Hong Kong, Osaka and Singapore. [26] Because of the surge or DDoS attacks in the region, the scrubbing capacity has increased at least three times to that of the largest DDoS attack. For all content delivery centres, traffic is redirected to scrubbing centres by making a BGP change. Upstreaming filtering or scrubbing is meant to prevent attacks before they reach data and applications in the cloud. The largest mitigated attack to date was Mirai at 1.3 TBPS. Distributed scrubbing centres aid in mitigation of such attacks.

Protocol Design Considerations: allow for forwarding of traffic on this scale through robust internet infrastructure. Attack traffic should be easily recognisable through its externals, e.g. packet destination, traffic flow patterns, protocol type, signatures. This relies on being able to filter at speed and weed out malicious connections.

4.8. Malicious Domain Monitoring and Takedown

Defence Type: Mitigation, detection and prevention

We wish to detect the existence and determine the intent of malicious domains as soon as possible, and remove or deny access to

them before most harm can be done. For example, removal of malicious domains that are created to receive clicks from phishing emails; if the domain can be removed before most emails are read, the links won't work and the harm is reduced with no effort from the user.

Takedown services can levy copyright protections to request takedowns. Combined with techniques to use email authentication, these proactive measures rather than reactive ones have had considerable effect in UK government efforts to minimise phishing. [27]

Case Study: TBD TODO

Protocol Design Considerations: Protocols should allow users to determine the identity of the domain that they are connecting before they are exposed to data from that domain. Protocols should provide a means for users to verify the authenticity of a connection to a domain. Protocols should minimise the opportunities for users to confuse malicious domains with legitimate domains. Protocols should provide a method for legitimate domain owners to recognize attempts by a malicious domain to masquerade as the legitimate domain.

4.9. Filtering

Defence Type: Prevention and mitigation.

Filtering of traffic can be done according to block lists of addresses, content types or signatures specified by malware threat feeds. Filtering can also be done using statistical and machine learning techniques, e.g. for spam. Filtering can prevent malware infection or mitigate it by stopping the further spread of malware.

Case Study: Telstra and Covid-19 Scams

The CEO of Telstra warned that Covid-19 malware scams are a boom to malware brokers and attackers. The rise and prevalence of Covid-19 scams in the first six months of 2020 is not surprising, but the Telstra CEO said clearly that they are strengthening their screening and filtering activities on their networks. [28]

Protocol Design Considerations: Protocols should allow for selective and/or minimally intrusive analysis of traffic in order to determine whether to allow data through the filter. Malware may try to shape malicious traffic to appear like benign traffic, so protocol specifications should minimise the opportunity for malicious payloads to masquerade as legitimate payloads. For example, encrypting all data so that it looks the same, then you're removing

any discriminating features that filtering systems could use to base their decisions on.

Protocols therefore should be designed with an awareness that hiding features that expose malicious traffic as malicious will enable malicious payload delivery, therefore it would be responsible to work out which, and preserve features that, would still allow effective detection.

4.10. Implementation of Trust

Defence Type: Prevention.

A trusted ecosystem is one in which a user or organization has a level of confidence in the security and reliability of the system. No ecosystem can ever be 100% secure, but trust is created when risk analysis and technical mitigations are in place.

For example, content is filtered so that data from non-trusted sources is filtered out before it arrives. DMARC/SPF to prevent phishing, secure authenticated log-in, PKI certificate validity on TLS connections is enforce. Updates and patch

Case Study: TBD TODO

Protocol Design Considerations: Protocols should be resilient and should not prevent informed users from opting into services that protected delivery of only trusted content. Protocols should allow for verification of data source and integrity. Protocols should include policies for handling and management of non-trusted data.

4.11. Endpoint Security

Defence Type: prevention, detection and mitigation.

Security hygiene, like regular patches to fix the latest discovered software vulnerabilities, form an important part of the security of any system. However, some endpoints are unable to maintain their own security and can introduce vulnerabilities themselves.

Case Study: Netgear

Netgear is one of many examples available relating to prevention, detection and mitigation of endpoints. In July 2020 Netgear released security advisories for a number of its routers, modems, gateways and extenders. [29] Firmware updates to patch the issue are linked to the corresponding network device.

Protocol Design Considerations: Consider whether the protocol data is available for inspection by the endpoint security solution. At what point in the protocol stack is protected data decrypted and can be analysed or blocked by the endpoint security?

4.12. Email Anti-spoofing Measures

Defence Type: Prevention

These are preventive measures designed to prevent and reduce phishing emails. Configure anti-spoofing controls on a domain you own to prevent email spoofing, such as Domain-based Message Authentication, Reporting and Conformance (DMARC), Sender Policy Framework (SPF) and Domain-Keys Identified Mail (DKIM). [30]

Case Study: Cosmic Lynx

Cosmic Lynx is a new business email compromise cybercrime gang which has already had over 200 targets in 46 countries since July 2019. [31] The criminal gang focus their attack on companies which don't deploy DMARC by using a fake 'mergers and acquisitions' email. The emails are linguistically well thought out and uses words not seen in phishing scams normally. Throughout the email correspondence about the business deal, they can directly spoof reply-to email addresses in order to look legitimate. Recipients are scammed into participating in sales and payments to attackers' bank accounts.

Protocol Design Considerations: Allow for strong authentication of source of user data. Create policies for delivery of non-authenticated data. Ensure authentication of all communication at the protocol level and enable security checks and communication at all stages of the process.

4.13. Social/User Interface Interactions

Defence Type: Prevention and detection.

Since phishing is a social engineering type of attack, there should be education and training for people to prevent phishing. Furthermore, presenting a user with a photo on log-in to prevent logging into a phishing domain and two factor authentication (2FA) are some of the mitigation strategies. Also reporting of abnormal behaviour/user mistakes should be encouraged and failed log-in attempts should be displayed to the user. Finally appropriate password policies should be in place.

Case Study: TBD TODO

Protocol Design Considerations: Protocols should support secure communication of security-critical information to and from the user interface; this could include passwords, biometric information, other credentials, user activity logs, PKI certificate properties and validity, origin authentication using auxiliary information (such as identifying phrases or photos).

4.14. Detection of Exfiltration and Data Leakage

Defence Type: detection and mitigation

When a compromise of a network has occurred, either by malware infection or insider threat, it is important to be able to detect attempts to exfiltrate data from the network and stop exfiltration as soon as the leak has been reliably confirmed.

Detection of exfiltration can be through packet metadata analysis, statistical analysis of data collected over a period of time, or content inspection on unencrypted data.

Case Study: SamSam is a ransomware that infects and then attacks after a period of monitoring networks and/or users. [31]Instead of attacking right away like WannaCry, SamSam manages to infect, delay and then attack again with the goal of infection, data infiltration and ransoming.[32]

Protocol Design Considerations: Encryption of data can make inspection of data at a gateway for malicious exfiltration less reliable.

Statistical properties of traffic may be used to detect exfiltration occurring over an extended period of time; it would be very bad for attack defence in general if protocols sought to hide patterns of traffic that are be indicative of exfiltration. If data is watermarked to indicate the origin of protected content, protocols should not destroy the watermarks. Protocols should minimise covert channels that can be used for the exfiltration of data by malware.

Additionally, designing a recurring monitoring and reporting mechanism within the protocol would allow for regular and consistent logging.

4.15. Misuse of the Domain Name System.

TODO

Defence Type: Detection and mitigation

Case Study: TBD TODO

4.16. Attack and Defense Table TODO

This section will be a table with attack, defence type, case studies, links and comments on the impact to protocol designs. This will be completed once the case studies have been added.

5. The Overall Security Picture

Deployments of protocols vary greatly and use cases show the variety and diversity of design, development and implementation of protocols; There are varying levels of risk and a variety of threats being more likely than others depending on context in which the protocol is deployed. Therefore, some attacks and defensive measures outlined in the above sections may be more frequent than others. For example, an enterprise might consider customer data exfiltration a greater threat than its resilience to zero-day vulnerability exploitation, but an individual user might be more concerned with their protection against phishing than with seeing all traffic leaving their network.

There is no one-size-fits-all approach for protocol deployment; each specific implementation and use case should be considered separately, as deployments require a mature a whole-system security view. This allows for a system wide analysis so that the security of the protocol isn't the only security considered.

For example, a user with a client that runs DoH might feel completely secure, as the information is encrypted and the user has a private connection to the DNS resolver. However, this could actually bypass defensive filtering protections, without the protection afforded blocking of malicious domains. Further, unless DNSSEC is also deployed, you have no trust that the resolver is returning the correct results and no passive auditor to check this.

Another popular example is the padlock in most browsers that tells users they have a secure HTTPS connection. Users can conflate the meaning of the padlock, assuming that use of HTTPS automatically confers a legitimate connection - even if the domain being connected to is fake or malicious.

6. Attack Defence in Security Considerations

The impact of new protocols on existing systems that defend against malicious attacks is not systematically considered in the Security

Considerations sections in RFCs. This draft is the first step in developing a reference guide to enable a systematic and consistent assessment across different protocols with respect to attack defence.

Hence, protocols should be assessed against a range of attacks and detections methods, such as those attack types listed in the Table in Section 2 and those defensive measures listed in the Table in Section 4, as a standard consideration in all protocol design, and to make the potential impacts clear to deployers.

When writing the Security Considerations for a protocol, protocol designers should consider known attacks and prevention, detection and mitigation methods. As the type, kind and characteristics of attacks grow in complexity, it is important that protocol designers take into account attack types and mitigation strategies into their designs. In fact this should be backed into the security considerations from the start. This draft RFC is a helpful guide to those considerations.

7.Security Considerations

This document is entirely about Internet security and is an input to the IAB Model T work.

8. IANA Considerations

This draft has no actions for IANA.

9. Conclusions

This draft is a work in progress, but is a set of considerations for protocol designers and implementors with respect to attack defence. Collaborations and contributions would expand this document to make it more robust.

10. References

10.1. Normative References

[1] RFC 7252 Shelby, Z. et al, "The Constrained Application Protocol (CAP)" RFC 7252, June 2014.

[2] https://www.zdnet.com/article/new-silex-malware-is-bricking-iot-devices-has-scary-plans/

[3] https://www.omdia.com/resources/product-content/maersk-ciso-offers-important-lessons-three-years-after-notpetya-attack-int005-000068]

[4] https://www.theverge.com/2016/10/21/13362354/dyn-dns-ddos-attack-cause-outage-status-explained

[5] https://www.zdnet.com/article/the-coap-protocol-is-the-next-big-thing-for-ddos-attacks/

[6] https://www.darkreading.com/endpoint/91--of-cyberattacks-start-with-a-phishing-email/d/d-id/1327704

[7] https://www.wired.com/2016/01/everything-we-know-about-ukraines-power-plant-hack/

[8] https://developers.facebook.com/docs/certificate-transparency/

[9] RFC 6962 Laurie B., et al, "Certificate Transparency" RFC 6962, June 2013.

[9] https://techcrunch.com/2019/05/12/wannacry-two-years-on/

[10] https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html#

[11] https://www.kaspersky.com/resource-center/threats/ransomware-wannacry

[12] https://www.cnbc.com/2017/07/31/new-anthem-data-breach-by-contractor-affects-more-than-18000-enrollees.html

[13] https://www.zdnet.com/article/for-two-hours-a-large-chunk-of-european-mobile-traffic-was-rerouted-through-china/

[14] https://www.internetsociety.org/blog/2018/04/amazons-route-53-bgp-hijack/

[15] https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends/enisa-threat-landscape

[16] https://www.welivesecurity.com/2020/01/09/mozilla-rushes-patch-firefox-zero-day/

[17] https://www.forbes.com/sites/emilsayegh/2020/02/12/more-cloud-more-hacks-pt-2/#7c0c47d669b3

[18] https://threatpost.com/google-patches-chrome-browser-zero-day-bug-under-attack/153216/

[19] https://www.crowdstrike.com/resources/reports/2020-crowdstrike-global-threat-report/]

[20] https://www.computerweekly.com/news/252446423/Industrial-controls-systems-a-specialised-cyber-target

[21] https://www.cnet.com/news/equifaxs-hack-one-year-later-a-look-back-at-how-it-happened-and-whats-changed/

[22] https://www.wired.com/2015/11/four-indicted-in-massive-jp-morgan-chase-hack/

[23] https://www.darkreading.com/risk/microsoft-hands-off-nitol-botnet-sinkhole-operation-to-chinese-cert/d/d-id/1138455]

[24] https://www.darkreading.com/risk/microsoft-hands-off-nitol-botnet-sinkhole-operation-to-chinese-cert/d/d-id/1138455]

[25] https://www.itproportal.com/features/how-it-can-combat-the-rise-in-vishing-attacks-in-this-new-normal/

[26] https://securitybrief.com.au/story/second-akamai-scrubbing-centre-opens-in-melbourne

[27] https://www.ncsc.gov.uk/guidance/phishing

[28] https://www.smh.com.au/business/companies/weakened-defences-covid-19-a-boon-for-malware-merchants-warns-telstra-boss-20200505-p54q2a.html]

[29] https://kb.netgear.com/000061982/Security-Advisory-for-Multiple-Vulnerabilities-on-Some-Routers-Mobile-Routers-Modems-Gateways-and-Extenders

[30] https://www.ncsc.gov.uk/collection/email-security-and-anti-spoofing

[31] https://threatpost.com/russian-bec-gang-cosmic-lynx-uncovered/157166/

[32] https://www.infradata.co.uk/resources/what-is-samsam-ransomware/

[33] https://blog.malwarebytes.com/cybercrime/2018/05/samsam-ransomware-need-know/

[34] https://www.informationsecuritybuzz.com/study-research/new-intelligence-reveals-that-alina-point-of-sale-malware-is-still-lurking-in-dns/

11. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Dominique Lazanski
Last Press Label
London, UK

Email: dml@lastpresslabel.com