CENTRIPETAL
NETWORKS

Intrusion Prevention is the Wrong Strategy for Malware Defense

A Technical White Paper
Centripetal Networks, Inc.

Intrusion Prevention is the Wrong Strategy for Malware Defense

Introduction

Malware attacks rival Denial-of-Service (DoS) attacks as the #1 cyber threat. In a malware attack, a resource is tricked into executing stealthy applications that cause damage, allowing theft or other misuse of the resource. Because the Internet is often the vector by which malware enters an enterprise's network and infects its resources, an obvious defense strategy is Intrusion Prevention: prevent malware from entering the enterprise network by inspecting all inbound Internet traffic.

In practice, intrusion prevention systems (IPS) and intrusion detection systems (IDS)[1] work by comparing inbound IP packets to a database of attack patterns, or signatures, of known malware. If a packet (or collection of related packets) matches a signature, then malware has been detected, and the associated packet(s) are blocked, i.e., the intrusion of the malware into the enterprise network has been prevented. This signature-matching approach had some early success, which resulted in a significant market in IPS/IDS-based solutions.

Unfortunately, however, these signature-matching systems have not been able to scale up to the threat. As malware signature databases grow larger in response to rapidly evolving malware attack patterns, an IPS needs increasingly more compute resources to process incoming traffic without causing unacceptable degradation of network performance[2]. Because adversaries can readily adapt their attack patterns to subvert databases of known signatures, they are overwhelmingly winning this cyber arms race, in terms of scale and adaptability. Regarding scale, Symantec reports detecting more than 400 million unique malware variants during 2011[3]. Regarding adaptability, many malwares are used only once, which means they are not detected by signature-matching systems and it is practically useless to develop and apply signatures for them. Because an IPS is limited to applying a few thousand signature-matching rules without incurring unacceptable degradation of network performance, the only strategy available is to select the subset of signatures of known malware capable of inflicting the most damage. The selection process is more skill than science. This strategy allows many known malwares, and all unknown malwares, to intrude.

Hence, as a cyber security strategy, intrusion prevention is, at best, a partial solution. Without some (unlikely) technological breakthrough, its effectiveness will continue to diminish relative to the malware threat. A new and different strategy is needed. First, consider that the malware intrusion event, by itself, does not cause any damage at all. Instead, damage occurs as a side effect of the malware being executed by some computing resource. Thus, one different general strategy would be to prevent the malware from executing or otherwise completing harmful operations. If this malware execution prevention strategy could be realized in practice, then it would potentially be much more effective and beneficial than intrusion prevention. Also, it is quite possible that an execution prevention solution may be a computationally simpler problem than intrusion prevention because theoreticians have proved that intrusion prevention is the most difficult type of computational problem to solve, and it is impossible to create a solution that is 100 percent effective[4]. Therefore, malware execution prevention cannot be any more difficult than intrusion prevention, and it may prove to be simpler and more effective.

Thus, since (a) intrusion prevention is not working in practice, (b) it does not directly counter the adverse effects of the attack, and (c) it is theoretically proven to be computationally difficult and not solvable, then it is logical to conclude:

A: Intrusion prevention is the wrong strategy for malware defense.

Of course, hindsight is 20/20. When intrusion prevention technologies first emerged as a significant improvement over network firewalls, the strategy appeared to be a good one. And today, there is certainly some value provided by IPS/IDS solutions and derivative technologies and processes, and they should continue to be part of the cyber security arsenal. However, it is clear that intrusion prevention technologies are not the panacea that the industry once hoped they would be. It is time to invest in a new strategy:

1. By definition, an IPS works inline on live traffic to prevent malware from entering a network, whereas an IDS works offline on stored traffic to detect when malware intrusions have occurred. Thus, an IPS has stringent performance requirements (low packet delay and packet loss) that can only be met by reducing the number of signatures applied to the traffic, i.e., reducing the provided security/protection. An IDS does not have stringent performance requirements and can apply much larger signature databases to detect malware intrusions; however, an IDS does nothing to prevent the damage caused by the malware it detects.
2. As measured by latency caused by packet processing delay, and packet loss caused by buffer overflows.
3. Symantec Corporation, "Internet Security Threat Report: 2011 Trends", Vol. 17, April 2012.
4. V. Sekar et al., "Network-Wide Deployment of Intrusion Detection and Prevention Systems", Proceedings of ACMCoNEXT, 2010.

B: A better strategy is malware execution prevention.

This is a new idea, and as such there are no marketready products and services available today that use this strategy – but it's time to start researching and developing solutions. Consider the three primary types of adverse effects caused by malware: (1) damaging a resource's assets, such as file erasures or corruptions; (2) misusing a resource, such as hijacking a control system; and (3) stealing sensitive data from a resource via the Internet, such as financial information, personal information and login credentials. In terms of security categories, the first two adverse effects types should be addressed mainly by host security (HOSTSEC) and information security (INFOSEC) measures, and the third by network security (NETSEC) measures. Because our interest is network security, in this paper we will explore the third one: stealing sensitive data over the Internet, a cybercrime known as exfiltration. Thus, we seek new approaches to:

C: Exfiltration Prevention:

Stopping exfiltrating malware from stealing sensitive data. Exfiltration is possibly the largest cyber threat to the United States. Gen. Keith Alexander, director of the NSA and commander of US Cyber Command, estimates that cyber-crime costs exceeded $1 trillion in 2008, and that costs have grown rapidly since then[5]. Furthermore, the exfiltration threat is highly asymmetric, meaning a few individuals with cheap, easily obtained tools can inflict massive damage and financial losses, and cause huge breaches in national security, even on strongly defended networks using the most sophisticated, expensive cyber defenses. An efficient, effective solution for exfiltration prevention would have a huge impact on US national security and the US economy. It would reverse the asymmetry by making it much more difficult for adversaries to steal data over the Internet.

CNI believes efficient, effective solutions for exfiltration prevention are feasible. This white paper explores some potential approaches to such solutions, after providing some relevant details on why conventional intrusion prevention technologies are not effective in stopping exfiltrations. The insights are presented in a pragmatic, accessible way intended for both non-experts and experts who want to understand why the effectiveness of intrusion prevention is rapidly diminishing, and conversely how exfiltration prevention could potentially be an asymmetry reversing strategy and what new technology advances are needed to realize practical solutions.

5. Gen. Keith Alexander, Military Information Technology, Vol. 14, No. 10, November 2010.

D: Exfiltrating Malware: Attack Models and Conventional Defenses

Exfiltrations are thefts of sensitive data via the Internet. Exfiltrations are perpetrated by:

(a) Malware: Exfiltrating malware is often surreptitiously downloaded onto a host that contains sensitive data (e.g., military secrets, financial account information, etc.), or a human operator uses it unknowingly to hand over sensitive data (e.g., account login information, PIN codes, etc.). The malware finds or collects the sensitive data and sends it over the Internet to collection servers, without the host owners/operators being aware of the theft. A popular type of exfiltrating malware is called a trojan (and is described below).

(b) Humans: Human operators may either intentionally or unintentionally (e.g., via e-mail phishing attacks) send sensitive data to collection servers.

The (False) Assumption of Insider Trust: Exploiting Firewalls and Intrusion Prevention Systems

At first glance, one may naively assume that conventional firewalls and IPS technology may be used to directly prevent exfiltrations, because both filter packets for suspicious or known threat information. Firewalls and IPSs are the state-of-the-art, de facto standard technologies for providing network security to enterprises and consumers.

Network firewalls are network devices that examine packets flowing across a network boundary, such as an enterprise's Internet access point. Firewalls either block or allow the packets according to packet filtering rules that, for performance and efficiency, examine only IP

packet header and transport (TCP/UDP/ICMP) packet header information. Thus, it would seem, at the very least, that firewalls could be used to block exfiltrations to known collection sites. There are two fundamental issues, however, that make this approach impractical: (1) the number of rules necessary to filter all known collection sites is too large for conventional firewalls to process; and (2) firewalls assume that insiders, including malware on hosts located inside their zone of coverage, can be trusted. This (False) Assumption of Insider Trust (described below) actually assists exfiltrating malware in perpetrating their crimes.

An IPS compares inbound packets to a database of attack patterns, or signatures, of known malware. The comparison typically examines information deep in the packet, e.g., the contents of application packets (e.g., HTTP packets that are encapsulated in IP and TCP packets), a technique called deep packet inspection (DPI). If a packet or a collection of related packets matches a signature, then malware has been detected and the associated packet(s) are blocked, i.e., the intrusion of the malware into the enterprise network has been prevented. Note that the use of commonly available encryption (e.g., used by the HTTPS protocol for encrypted web sessions

makes it essentially impossible for an IPS to inspect packets, and is therefore a very simple way for adversaries to subvert an IPS.

Firewalls and IPSs are often used in tandem, with the firewall being placed in front of the IPS, i.e., the firewall filters inbound Internet traffic before it is sent to the IPS. This is efficient because the firewall can filter unsolicited traffic destined for non-public resources much more cheaply than the IPS[6]. The way that the firewall/IPS tandem handles insiders' outbound traffic and the associated inbound solicited traffic, however, is readily exploited by adversaries and malware to perpetrate exfiltrations, as follows (refer to Figure 1 to the right):

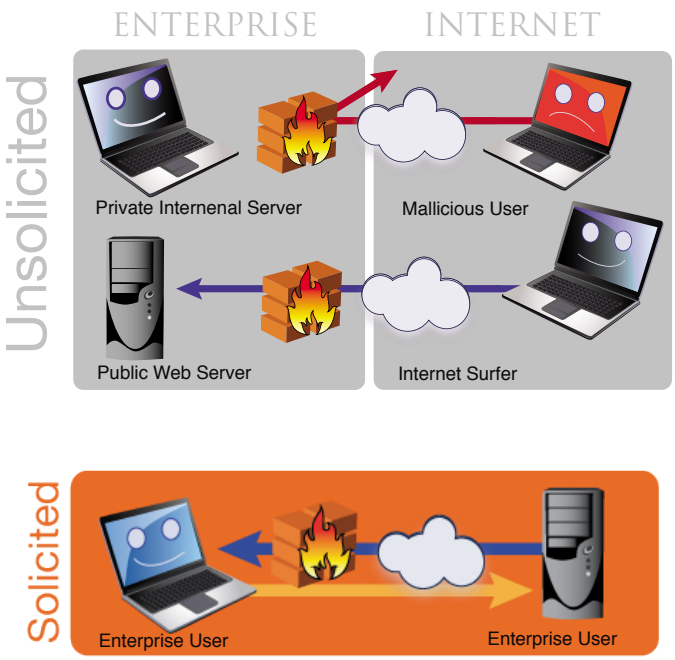


Figure 1: Solicited and Unsolicited traffic between the Internet and an Enterprise using a firewall/IPS tandem to protect resources and users

Firewalls and IPSs distinguish between inbound and outbound traffic, and solicited and unsolicited[7] traffic. Firewalls readily block unsolicited inbound packets that are destined for protected resources (e.g., a private host/desktop computer, private enterprise data center, or enterprise network) located behind the firewall, i.e., on the protected side of the network security boundary. But a conventional firewall's trust model assumes that any session initiated by a resource located behind the firewall – an insider – can be trusted, so the firewall by default allows all outbound traffic[8]. An IPS, by definition, does not inspect outbound traffic[9], so by default it assumes the firewall's trust model. Thus, generally speaking, all outbound traffic is allowed because it is trusted.

6. Some IPS devices include a firewall, which in effect is the same as the tandem configuration.

7. Solicited traffic is inbound traffic that is a response to a request made by a hosted application behind the firewall. For example, when a web browser behind the firewall initiates a web session by requesting (soliciting) a web page from a web server, the packets containing the web page information, and sent by the web server to the web browser, is considered to be solicited traffic by the firewall. Unsolicited traffic is inbound traffic that is initiated by an application hosted outside the firewall and destined for a resource behind the firewall; for example, a web browser request for the home page of an enterprise's public web server located behind the enterprise's firewall.

8. A notable exception is outbound traffic destined for an Internet location that the enterprise does not allow communications with, e.g., a pornography site, or a known cyber crime site, such as the Russian Business Network. Which begs the question, why don't enterprises configure their firewalls to block all outbound traffic to all known cyber crime sites? Because the number of rules necessary to block all such sites far exceeds the capability of conventional firewalls.

9.Of course, an IPS can inspect outbound traffic, but it is extremely difficult to define signatures that discriminate between insider traffic generated by legitimate users vs. insider traffic generated by malware and malicious insiders. In general, therefore, an IPS is not used to inspect outbound traffic.

This is the (False) Assumption of Insider Trust.

Inbound packets that have been solicited by a protected resource located behind the firewall – such as packets containing a web page requested by a web browser – are allowed to cross the security boundary. Firewalls implement this trust model by maintaining state information on Internet sessions initiated by resources located behind the firewall; accordingly, these firewalls are characterized as stateful firewalls. In a nutshell, stateful firewalls examine outbound packets and record the source and destination IP address and the source and destination port (values located in the packets' headers). Then, an inbound packet with source/destination IP addresses and ports that match an outbound packet's destination/source IP addresses and ports is allowed to cross the boundary.

Finally a firewall allows unsolicited inbound traffic if it is destined for a publicly available/addressable resource, such as a public web server. Any solicited or unsolicited inbound traffic that is allowed by the firewall is then filtered by the IPS for malware.

How Malware Exploits the (False) Assumption of Insider Trust

Exfiltrating malware exploits the (False) Assumption of Insider Trust to perpetrate its crimes. The Assumption makes it trivial for malware to steal sensitive data. Figures 2 and 3 below illustrate how trojan malware steals data, and similarly how spear-phishing e-mail steals data.

Exfiltrations by trojans[10] are typically perpetrated as illustrated above in Figure 2 and as described below:

Step 1: A link to a malware server is inserted, via infection, into a web page of an otherwise-legitimate web server.

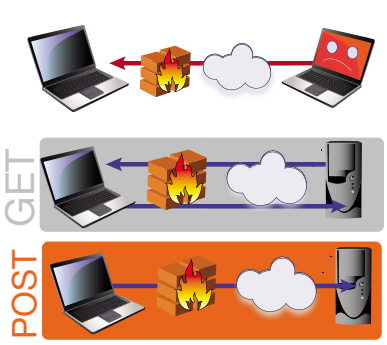
Step 2: The user who downloads the web page is enticed to click on the link[11] to the malware server. This initiates a download from the malware server (the firewall/IPS assumes insider-initiated communications are trusted, so it allows the download). The malware is installed on the user's host machine, and begins collecting sensitive data (e.g., usernames and passwords to user's online bank accounts) or locating locally stored files containing sensitive data.

Step 3: The malware then exfiltrates the data and/or files by sending them to a collection server. Again, the firewall allows the exfiltration since the malware appears to be a trusted insider. Some malware even encrypts the exfiltration content (using, e.g., TLS and HTTPS) to avoid detection by deep-packet-inspection (DPI) systems, such as IPS solutions.

Phishing attacks/exfiltrations are similar in structure to trojan malware attacks/exfiltrations. In a typical spear phishing attack (illustrated below in Figure 3), the victim is enticed to open an e-mail attachment – which actually launches a malware application – or is enticed to click on a link in an e-mail which appears to be a legitimate request from another user known to the victim, or from a business organization, e.g., a bank, known to the victim. The link takes the victim to a web site, which either downloads malware or tricks the victim into divulging sensitive information (e.g., bank account login credentials). In any case, the last step in a phishing attack is the same as Step 3 of the trojan attack described above.

10. For an in-depth, accessible description of a typical (and highly "successful") exfiltrating trojan called Torpig, see "Analysis of a Botnet Takeover", IEEE Security & Privacy Magazine, January/February 2011.

11. Some enterprises use "URL blacklisting" systems that compare the URL/domain of outbound web page requests with a list of known malicious web sites and malware servers. Malware easily defeats these systems by encoding the IP address of the malware server in the malicious link instead of the server's URL.



1. E-mail sent by attacker to phishing victim. E-mail includes web link to malicious web site, which appears to be legitimate. Firewall/IPS not applied to email.
2. Victim enticed to click on link, which brings victim to "legitimate" web site familiar to victim (e.g. online banking). Victim enters SSN, username/password, etc. into a web form. Firewall/IPS bypassed using Assumption of Insider Trust.
3. Victim hits "Enter" (posts the web form). Firewall/IPS allows POST using Assumption of Insider Trust. Attacker now has valuable data to sell or exploit.

Figure 3: Life Cycle of an E-mail Phishing Exfiltration Attack

Finally, human user, aka an insider attack, may perpetrate exfiltrations deliberately. A human user will use some data transfer mechanism, e.g., posting files on a website, instant messaging attachments, e-mail attachments, etc., to transfer sensitive data to a collection site.

How Did Cyber Security Become so Ineffective Against Exfiltrating Malware?

In both of the above examples – a trojan exfiltration and a spear-phishing exfiltration – one potential mechanism for stopping the exfiltration is to use the firewall to block any outbound traffic to known malicious sites (e.g., collection servers, malware-infected web sites, hosts known to be used by cyber criminals such as the Russian Business Network). This is a good idea and should be highly effective. In today's cyber environment, however, it is not practical, as follows.

Traditional firewalls filter on the so-called 5-tuple packet header fields: source and destination IP address, source and destination port, and protocol type (IPv4) or next header (IPv6). Traditional firewalls were effective when

(a) the sources of malware and the destinations of exfiltrations were few, thereby limiting in practice the number of filter rules to a few hundred, or at most a few thousand, rules; and when

(b) applications only used the well-known ports standardized by IANA (e.g., port 80 for HTTP web servers).

As the cyber threat grew in size and sophistication, however, the number of filter rules necessary to provide effective protections also grew past the ability of firewalls to apply them while maintaining sufficient network performance. For example, firewalls and router access control lists (ACLs)[12]typically enforce network security policies composed of hundreds, thousands or (infrequently) tens of thousands of rules, whereas at any given time there are several hundreds of thousands, or even a few million, known bad IP addresses which should be filtered to provide networks with comprehensive protections from Internet attacks. Also, since there is no capability in the TCP/IP protocols nor in host operating systems to enforce standard port usage, malware can easily subvert a firewall's port filtering rules.

Adversaries have responded to DPI methods in at least three ways:

1. Diversity: Continually creating new malware and attacks. Symantec reports the identification of more than 400 million unique malware variants in 2011. DPI-based systems applying signature matching cannot scale to the size of the threat;

2. Adaptability: Malware and associated attacks are continually modified so that yesterday's signature databases no longer match today's attacks and malware; and

3. Encryption: Intrusion and exfiltration sessions may be encrypted, thereby thwarting any DPIbased prevention methods that examine content. The popularity and ubiquity of the Transport Layer Security (TLS) protocol – which puts the "S" in HTTPS – for encrypting packet content has made it simple and cheap for attackers to conceal intrusions and exfiltrations.

Malware diversity/size, adaptability (polymorphism) and encryption have made DPI-based intrusion/exfiltration prevention ineffective, inefficient, or both.

Moreover, there is strong empirical evidence that the malware threat has overwhelmed the defensive capability of even the most sophisticated and highly scaled in-

12. Access control lists (ACLs) are packet filters that routers apply to their network interfaces. Like firewalls, ACLs filter on the 5-tuple of header values. Unlike firewalls, ACLs are not intended to be used as enterprise firewalls so they do not, for example, have a concept of inbound and outbound traffic, nor a concept of solicited and unsolicited traffic; and, therefore, do not maintain state in order to, e.g., allow solicited inbound traffic to cross the security boundary.

trusion prevention systems. For example, analysis of a recent US Government program called EINSTEIN 3[13] surfaced significant technical as well as policy concerns. Areas of technical concern include scale, fast correlation ability (quickly recognizing new threats), device management, signature management, and the need for man-in-the middle decryption. Areas of policy concern include privacy issues, what to do about encrypted traffic, massive data storage requirements, potential misuse and abuse of the system's devices, and cost.

Finally, privacy alone is a very significant societal issue. Cyber security solutions that routinely inspect content on public networks are unacceptable to public users and their political representatives.

Examples of solutions and legislation that can monitor public voice and data communications include CALEA and the Carnivore security solution, SOPA, PIPA, PCIPA, and EINSTEIN 3. Any general solution to the exfiltration problem needs to be privacy preserving.

Strategies for Exfiltration Prevention

It is clear that as a general strategy for defense against exfiltrating malware, intrusion prevention is not working. It's time to invest in a new strategy. **Exfiltration prevention** should be that new strategy. As illustrated in the trojan and spear-phishing exfiltration examples above, the malware intrusion event by itself does not cause the damage. The exfiltration event causes the damage. Thought leaders in cyber security are now promoting a strategy that assumes malware infection cannot be totally prevented, and therefore emphasis should be placed on mitigating or neutralizing malware's effects, which in the case of exfiltrations means exfiltration prevention.

Emerging exfiltration prevention research has been focused on content analysis and (closely related) behavior analysis. The concept is to be able to characterize normal, legitimate behavior of hosts, and then be able to detect exfiltrations – in "real-time", no less – as anomalous behavior. This approach is analogous to the signature analysis approach used in intrusion prevention, and it may be as difficult, complex, and compute intensive. A recent thesis on exfiltration methods shows that the data transfer protocols and methods used by exfiltrations are nearly identical to those used by legitimate functions[14]. If behavior analysis proves to be as complex, if not more so, as signature analysis, then behavioral analysis methods may also prove to be as ineffective and inefficient as intrusion prevention. If this is the case, then hopefully it will be realized early on so that R&D can be focused on more promising strategies and technologies.

CNI believes there are different, more effective methods, besides behavioral analysis, that may be applied to exfiltration prevention. Let us explore them, and also explore the types of technologies needed to implement them.

The emerging behavioral analysis methods and the intrusion prevention methods use DPI-based techniques which focus on packet content analysis – "what" contents the packets contain – to detect exfiltrations and intrusions. By itself, this single-dimensional approach cannot scale to the size and complexity of the threat. Moore's Law – in the form of applying faster and more cost-efficient compute resources to packet content analysis – will not help. Not only is the threat is growing faster than the speed/cost ratio of processors, but also even the most powerful (expensive) DPI-based solutions are easily and cheaply defeated by encryption (e.g., TLS). Alternative approaches that are not based solely on content analysis and behavior analysis must be considered.

5-tuple Filtering for Exfiltration Prevention: Cyber Enclaves

A first area to look for alternatives is 5-tuple filtering. Recall that the 5-tuple is comprised of source and destination IP address, source and destination port, and protocol type. These five packet header fields identify which network-attached hosts (IP addresses) and which application instances (ports) on those hosts, are communicating via a given packet. If packet content characterizes the "what" dimension, then 5-tuple filtering characterizes the "who" dimension.

13. S. Bellovin et al., "Can It Really Work? Problems with Extending EINSTEIN 3 to Critical Infrastructure", Harvard National Security Journal, Vol. 3 (2011), pps. 1-38.

14 R. Van Antwerp, "Exfiltration Techniques: An Examination and Emulation", University ofDelaware Library, 2011.

As noted above, 5-tuple filtering used in conventional firewalls was de-emphasized by the cyber security industry during the last epoch of technology evolution[15]. In a nutshell, the reasons for this de-emphasis were that: (1), the size of the threat, – as measured by the number of Internet-attached hosts (IP addresses) used in attacks, – has far outgrown the packet filtering capabilities of conventional firewalls and router ACLs; and (2), the inability of enterprises to enforce standard usage of ports. For example, very few organizations actually enforce network security policies in which networked applications only use their IANA-assigned ports for communication. Note that the standard-port-enforcement reason number two is highly correlated with the scalability issue of reason number one, because packet filtering at the fine granularity of {IP address, port} pairs increases the number of filtering rules. Note also that malware often use nonstandard ports to avoid detection and to avoid conflicts with legitimate applications using the standard ports. Finally, as noted above, hundreds of thousands of malware sites are well known to the cyber defense community, but this is far too many sites for firewalls and router ACLs to block.

One potential way to address the scalability problem is to use a converse approach. Instead of trying to block all of the communications/packets coming from or going to known malware sites, this approach only allows communications/packets between legitimate, approved network resources. For example, a geographically distributed, multi-site enterprise could enforce network security policies that allow inter-site communications only between specific pairs of enterprise resources. Such a policy is called an "allow list", since it contains packet-filtering rules that allow a packet only if it matches one of the rules.

More specifically, suppose an Enterprise operates three (network) sites: Site A, B, and C, interconnected by the Internet. A packet filtering device located at Site A's Internet access point only allows outbound packets destined for hosts (IP addresses) located in Site B or C; and, the device only allows inbound packets sourced by hosts located in Site B or C. All other packets are blocked, including all attack packets originating from arbitrary hosts attached to the Internet. The packet filtering devices located at Site B and C have similar allow-list policies that restrict packet communications to sessions between the Enterprise's resources.

Note that the example above uses a granularity of IP addresses or individual host machines. The granularity can be increased to individual application instances on specific machines by specifying port values in the rules.

For example, an internal web server accepts requests on IANA-standard port 80 (HTTP) and port 443 (HTTPS). Most of the Enterprise's users (and their machines) should only be communicating with the web server's host machine on port 80 and/or port 443. Attempts to communicate on other web server ports, e.g., port 22 for SSH (used for administrative logins to manage the web server system), by unauthorized users/machines should not be allowed, as this is potentially a malware attack or an insider threat. The allow list policies should therefore have port-level granularity. Conversely, in some cases it may be more efficient to have less granularity than individual IP addresses, in which case subnets should be specified in the allow lists.

We say that a set of networked resources is a "cyber enclave" if communications between the resources is closed; i.e., resources in the cyber enclave can only communicate with other resources in the same cyber enclave, and cannot communicate with resources that are not in the enclave. In the above example, the hosts attached to the networks of Site A, B, and C form a cyber enclave. Internet-attached hosts, including known malware sites, are not in the cyber enclave. Note that the allow-list policies enforced by the packet filtering devices define the cyber enclave, because they define which resources can inter-communicate.

At first glance, the cyber enclave approach appears to defend against not only the exfiltration prevention problem, but also many other cyber threats. Why hasn't the cyber enclave approach been deployed widely? There are at least three issues preventing deployment:

1. Internet Access: Except for "air-gapped" networks that may be found in the military and intelligence communities, enterprise users will require access to many Internet-attached hosts to perform their jobs. Not only is this potentially a huge number of hosts (e.g., IP addresses of servers hosting websites) which will drive up the number of rules, but also many users will not know, in advance,

15. Internet Access: Providing Internet Access outside the closed cyber enclave, while simultaneously preventing exfiltrations, will require some new approaches and technologies. Solutions to the Scale and Automated Tools issues are necessary, but not sufficient, to solve the Internet Access issue.

which Internet-attached hosts they will need to access to perform their job functions. Unless some method is employed to control the number of rules that allow appropriately unrestricted access to the Internet, the cyber enclave may not be feasible.

2. Scale: Regardless of the number of rules needed to provide Internet Access (above), the number of cyber enclave rules needed for just the intra-enterprise communications will likely exceed the capabilities of conventional firewalls and router ACLs. For example, if an enterprise has several thousand resources, and granularity is specified to individual applications instances, then it is expected that the associated cyber enclave may have several hundred thousand or even a few million rules. This is because, in general, a cyber enclave's rules specify communications between pairs of specific resources, and the combinatorics of pairs grows quadratically, as follows: if there are N resources (e.g., 1000 hosts), then an associated cyber enclave will have on the order of N2 rules (1 million rules). Although clever rule-reduction methods should significantly reduce the size of cyber-enclave rule bases to some fraction of N2, the number of rules will still be significantly larger than the number of resources. The most powerful (and expensive) conventional firewalls and router ACLs are limited by performance to applying network security policies composed of several thousand rules, and thus they cannot scale to the needs of cyber enclaves.

3. Automated Tools: Even if a filtering technology is capable of applying policies composed of millions of rules, efficient automated tools will be necessary to create and manage cyber enclave policies[16]. Without efficient automated tools, human network operators cannot possibly manage cyber enclaves.

Automated Tools: Of the three issues, this one is probably the simplest one to address, but it is still non-trivial. CNI has developed a preliminary design for efficient automated tools, which may be used to generate cyber enclaves. Also, there is at least one emerging cyber-security technology project, which automatically generates cyber enclaves from user and host information[17], although the efficiency and scalability of the process is not specified. Such tools do not yet exist because the cyber enclave concept and method is new. As the idea gains traction and deployments occur, it is likely that highly efficient automated tools will emerge.

Scale: To solve this problem, new efficient packet filtering technology is required. Performance improvements, as measured by the size of rule databases, latency, and packet loss, must be significant, as much as 1000X or more. Moore's Law – processor performance doubles approximately every 18 months – will not be of much help[18]. Instead, algorithmic breakthroughs will be necessary. Theoretically speaking, packet filtering algorithms have not improved since the time packet filtering was first used. But as with Automated Tools, CNI is aware of recent advances in packet-filtering algorithms, and new products which improve performance by several orders of magnitude. Internet Access: Providing Internet Access outside the closed cyber enclave, while simultaneously preventing exfiltrations, will require some new approaches and technologies. Solutions to the Scale and Automated Tools issues are necessary, but not sufficient, to solve the Internet Access issue.

Internet Access: Providing Internet Access outside the closed cyber enclave, while simultaneously preventing exfiltrations, will require some new approaches and technologies. Solutions to the Scale and Automated Tools issues are necessary, but not sufficient, to solve the Internet Access issue.

Regarding **Automated Tools**, cyber enclave policy creators will, in general, know their networks' IP address spaces, the characteristics of their resources, and the communications policies between their resources. But, they cannot possibly know and manage this information for the Internet, in such a way that they can extend their cyber enclave policies to arbitrary resources attached to the Internet and thereby allow their enterprise users to safely and securely access the Internet, surf the web, etc. Regarding **Scale**, the conventional approach of using DPI-based content analysis and behavior-analysis methods to prevent exfiltrations does not scale to the

16. A definition of efficiency in this context is as follows: Assume that there are N hosts (IP addresses) or N networks (subnet prefixes) or even N application instances (IP address and port) that are included in the resource pool to be protected by the enclave. Then, efficient automated tools are able to automatically generate cyber enclave policies with on the order of N2 rules while requiring the human operator to manually manage only N information chunks and perform only N steps to generate the cyber enclave policy. In concrete terms, if N = 1000 (hosts), then N2 = 1,000,000 (rules).

17. G. Nakamoto et al., "Identity-Based Internet Protocol Networking", IEEE MILCOM 2012, Orlando, FL, USA, October 2012.

18. To achieve a 1000X performance improvement with only Moore's Law will take about 15 years – and by then the threat will have grown, network links speeds will have increased significantly, etc., which makes this approach non-feasible.

size of the Internet threat. Other problems include cost, complexity, privacy issues, encryption, etc.

Q1: Who are the communicating entities, and are they allowed to communicate?
Q2: What is the type of the communication – read, write (potential exfiltration), encrypted (potential exfiltration) – and is it allowed?

If the answer is "No" to either Q1 or Q2, then the communication should not be allowed because it does not follow the communications policy of the governing enterprise and is potentially an exfiltration. For example, an enterprise resource (e.g., a desktop computer) may be allowed to surf an Internet-attached web site (accomplished using the HTTP GET method), but will not be allowed to transfer files or transfer form data to the web site (accomplished using the HTTP PUT and POST methods, respectively), as these data transfers are potentially exfiltrations being perpetrated by an insider or by malware which has infected the enterprise resource. At first glance, this capability may appear to be mundane, but it cannot be accomplished using only conventional 5-tuple packet filtering. And, while it may be possible to accomplish it using DPI-based technologies, conventional implementations of DPI technologies are too inefficient to be used in practice at scale and during heavy loading.

Is there some middle ground between 5-tuple filtering and full DPI that balances processing requirements with the information requirements necessary to determine the type of data transfer operation? The answer is, potentially, "Yes".

For the data transfer protocols used by exfiltrations, the data transfer method/type can be determined by examining only application packet header information, without examining application packet content. For example, as shown above in Figure 4, an HTTP (web) application packet includes the transfer operation method – GET (surf), PUT (upload a file to a web server), POST (upload form data such as account credentials to a web server), etc. – in the header. The logic and processing resources necessary for examining and interpreting application packet header information are much simpler than those for examining and interpreting content. It is more work than filtering on just the 5-tuple, but considerably less work than fully analyzing content. Now, although this technique of determining data transfer type by only examine application packet header information is much cheaper than full DPI, it must be combined with "who" dimension information – 5-tuple cyber enclaves – to create an exfiltration prevention solution. Thus, the Scale issue remains on the critical path to yielding a practical exfiltration prevention technique.

But, how do we know if the "who" and "type" approach to exfiltration prevention will be any less complex and resource intensive than content analysis and behavior analysis approaches? The following hypothetical example suggests that the scope of an Internet-scale "who" and "type" exfiltration prevention solution may be smaller by orders of magnitude (1000X or more) than solutions based on content analysis and behavior analysis. As a rough measure of scope, we noted above that as of 2011, Symantec identified over 400 million unique variants of malware. Thus, any content-analysis and behavior-analysis systems using signature matching cannot be expected to deal with more than a tiny fraction of the total number of variants.

Now consider a hypothetical "who"-and-"type" exfiltration prevention system, named Exfil-Blocker. ExFil-Blocker has a filtering capability which not only can apply 5-tuple filtering rules to packets and but also can efficiently determine the data transfer method type (if any) of the application that sourced packet. Without loss of generality, we will consider only the HTTP protocol (which happens to be the most popular protocol for exfiltrations), and within HTTP, we will only consider the three (3) methods used for data transfers: GET, PUT, and POST. GET is used to request web pages from web servers, PUT is used to upload files to web servers, and POST is used to transfer form data – such as login credentials – to web servers. Note that exfiltrating malwares use PUT and POST to transfer sensitive data from the victim to the collection point.

A US-based enterprise, the XYZ Company, wants to use ExFil-Blocker to stop its intellectual property from being stolen by foreign governments. XYZ also wants to protect its employees' business accounts and personal accounts from phishing e-mail attacks. But, XYZ also wants to allow its employees to freely surf the web to conduct research in their respective business functions. And, XYZ needs to conduct business over the web with many US-based and UK-based companies. Finally, XYZ has a policy of not conducting any business over the web with any ITAR countries[19].

19.The US State Department's International Traffic in Arms Regulations (ITAR) control the export and import of defense-related products and services. The State Dept. maintains a list of countries, colloquially know as the "ITAR countries",

Thus, XYZ's Exfil-Blocker needs to enforce a network security policy that filters packets according to the country of origin or destination, i.e., according to the packets' "who" dimension values. This requires Internet geolocation data, which is a mapping of IP addresses an subnet prefixes to countries. There are both open and private sources of geo-location data. One open source organizes the geo-location data according to subnets[20] assigned to countries. There are approximately 175,000 IPv4 subnet prefixes in the geo-location database, and thus there needs to be at least this many rules in Exfil-Blocker's network security policy. Because of the network security requirements, the network security policy will need to filter on both source IP addresses and separately on destination IP addresses, which means that at most there needs to be twice as many rules as subnet prefixes, or an upper limit of approximately 350,000 filter rules. In effect, these 350,000 rules are a measure of the scope of the problem. This is at least three orders of magnitude, or 1000X, smaller than the scope of the content analysis and behavior analysis problem, which has a lower bound of 400 million.

XYZ configures its ExFil-Blocker's network security policy with different types of rules, as follows:

- Approximately 15,000 packet filtering rules that block any packet, which originates from or is destined to a subnet in an ITAR country;

- Approximately 85,000 packet filtering rules that allow any packet which originates from or is destined to a subnet in the US or the UK, and which has type GET, PUT, or POST;

- Approximately 250,000 packet filtering rules that allow any packet which originates from or is destined to a subnet not in the US, UK, or an ITAR country, and which has type GET. Packets of type PUT or POST are blocked, as these are potentially exfiltrations.

This network security policy meets the requirements of allowing XYZ's employees to freely web surf to any web site located anywhere in the world except in ITAR countries, but only allows web-based data transfers to US and UK web sites. Any exfiltration attempts by malware, phishing attacks, and malicious insiders to non-US or non-UK countries, are blocked.

To recapitulate, the above arguments suggest that an exfiltration prevention solution that blocks malware from executing data transfers is probably a more solvable problem than solutions based on content analysis and behavior analysis. One reason is that the problem scope for the "exfiltration blocking" approach appears to be smaller, by orders-of-magnitude, than the scope of content/behavior analysis solutions. However, a break-through in the performance of packet filtering technology is needed before an exfiltration blocking solution can scale to the Internet.

Conclusion

Because intrusion prevention has failed as a general strategy for defending against malware attacks, it is time to explore new strategies. Where do we look for new strategies? The observation that a malware intrusion event, by itself, does not cause any damage, exposes the indirect nature of the intrusion prevention strategy. We should look for a direct strategy. Because the execution of the malware causes the damage, a direct cyber defense strategy is malware execution prevention. Therefore, for the case of exfiltrating malware, an exfiltration prevention strategy should be pursued. One realization of the strategy is a system that will prevent malware from transmitting sensitive data and credentials over the Internet to a collection site.

The cyber security research community is beginning to realize that exfiltration prevention is the right strategy. However, current state-of-the-art approaches are based on behavior analysis and content analysis. These approaches are very similar to those used in intrusion prevention, and therefore they are likely to be as difficult and complex, and therefore likely to fail.

CNI believes the fundamental problem with these current approaches is that they examine only one dimension of the exfiltration problem: the "what" dimension, i.e., they analyze content and behavior of packet traffic. The "what" dimension is extremely large in scope. The behavior of the malware when it is executed will vary similarly and will have similar problem scope. Analyzing malware

which have embargoes in effect for these defense-related products and services. The current list can be found at www.pmddtc.state.gov/embargoed_countries/index.html

20. In this context, a subnet is a contiguous range of IP addresses. IANA and Regional Internet Registry (RIR) organizations allocate IP addresses as subnets.

content and execution behavior by signature matching is a difficult, complex problem. The scope of the “what” dimension appears to be practically unbounded.

CNI proposes that exploiting two other dimensions of the problem – the “who” and the data transfer method “type” dimension – may lead to effective and efficient solutions for exfiltration prevention. When applied to exfiltration prevention, the “who” dimension factors in the Internet identities of the communicating resources. The “type” dimension factors in the data transfer operation – read, write, delete, etc. The two dimensions can be logically combined to produce an exfiltration prevention operator that is applied to any data transfer session that is about to occur. The operator first determines if the two communicating entities are allowed to communicate, and if so, then the operator determines if the type of communication – read, write, encrypted, etc. – is allowed between the two entities.

A hypothetical example of an Internet-scale exfiltration prevention system – called ExFil-Blocker – that applies the “who” and “type ” operators was shown to have a scope that is smaller by several orders-of-magnitude than the scope of conventional signature-matching systems. However, ExFil-Blocker depends on packet filtering technology with performance requirements that exceed the capabilities of conventional filters. Sufficiently powerful filtering technology is required to realize an effective exfiltration prevention system such as ExFil-Blocker. In contrast, it is highly unlikely that a new signature-matching technology will emerge which will make conventional solutions for exfiltration prevention effective and efficient.

Thus, we conclude that the cyber security industry should invest in the packet filtering technologies necessary to implement an Internet-scale version of the ExFil-Blocker exfiltration prevention system. Existing methods will fail to solve the problem, just as similar methods applied to malware intrusion prevention failed to solve the problem.