VECTRA®
Threat Labs

How to interpret network-based malware detection

The impact of malware acquisition and processing on network detection and threat classification systems

By Günter Ollmann
Chief Security Officer
Vectra Networks

VECTRA®
Threat Labs

TABLE OF CONTENTS

Despite almost four decades of anti-malware defense research and a continuous progression of detection and mitigation technology, malware reigns supreme at number one as the most ubiquitous information technology threat facing business and residential systems.

The unremitting arms war between malware author and security vendor has driven innovation on both sides – with each advancement mitigated in part by a corresponding technology advancement on the other side. It is a war bounded by economics rather than attrition.

As one door closes, malware authors seek the next lowest-hanging-fruit vector for deceiving or compromising a victim's computer. Meanwhile, vendors strive to invent and deploy detection advances that mitigate the most frequently encountered versions of the threat, and any permutation of the threat specifically affecting their largest customers.

Over the past decade, the primary malware barrier has moved from host protection to network detection. While host-based defenses provide the most advanced suite of anti-malware defenses and clean-up capabilities, they are also the most difficult to manage and often the easiest to circumvent.

Network-based anti-malware defenses – ranging from dynamic analysis sandboxing through to network traffic analysis – are typically seen as casting a wider and more economical net for detecting malware payloads and communications.

While network defenses have advanced against the malware threat, there is only so much that can be accomplished at that layer. Network-based detection of malware is based upon the network artifacts and communication features of the malware itself.

Consequently, for new defenses to be constructed, security vendors and researchers must typically have access to families of malware samples – whether their core technology is based on signatures, behavioral analysis or machine learning.

It is generally assumed that once a new malware sample is encountered that there is a linear progression of research that leads to an update to a threat detection system (e.g., a new signature or classifier). Unfortunately, there are many nuances to malware construction, agent deployment, and the supporting threat ecosystem that greatly limit what can be logically inferred from captured samples and converted in to actionable intelligence.

This e-book examines the ecosystem nuances upon network-based malware detection and the limits imposed on intelligence extraction of captured malware samples, and what the subsequent implications are upon organizations seeking to mitigate the malware threat based upon network-based detection systems.

The malware ecosystem

Gone are the days that a sole-proprietor malware author was your cyber-crime adversary. Today, every operational component of malware creation, distribution, control and monetization can be outsourced to dedicated professional entities around the globe. As a consequence, the prospect of threat attribution and takedown is becoming more elusive by the day.

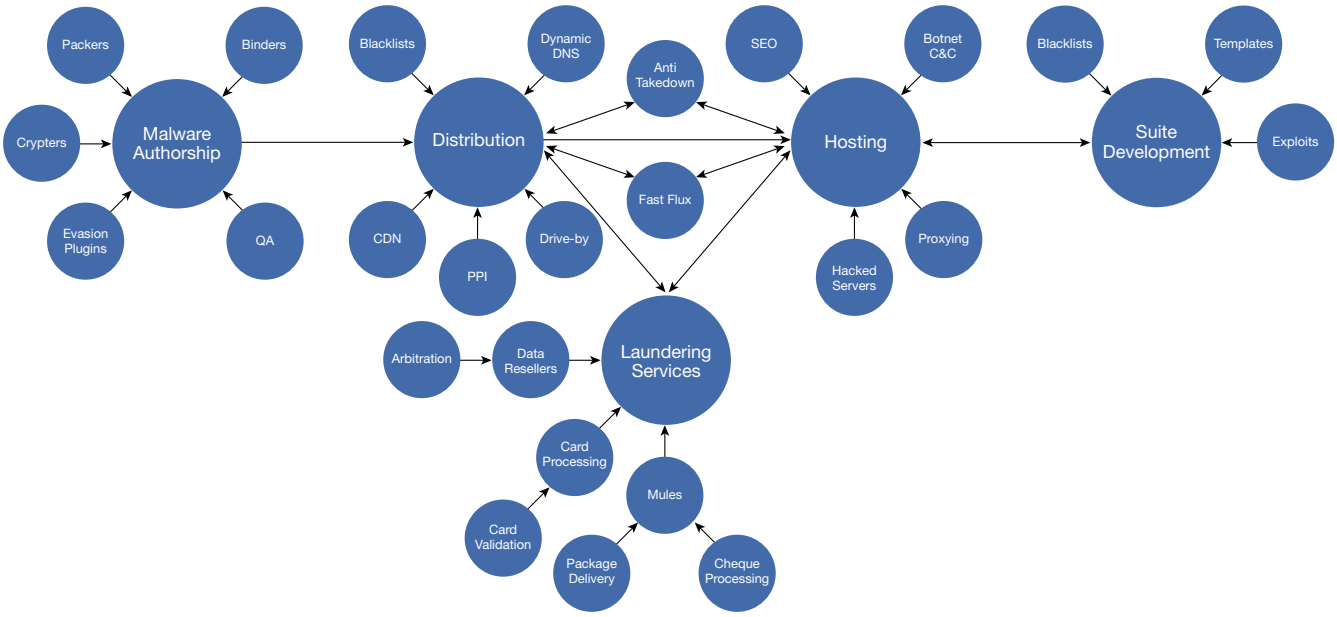

Figure 1: Simple malware ecosystem example.

To illustrate the complexity and interconnection of services that contribute to the malware ecosystem, some of the most common elements of today's ecosystem include:

Malware authorship

Malware authorship is the construction of the core malware code and binary packaging. This may be the dropper – a small package designed to download and install a different malicious file – or the full malware payload (e.g., a botnet agent).

- Packers – Tools designed to compress malware payloads. Most packers include capabilities to thwart static analysis, file heuristics, and emulator-based analysis of packed files.
- Crypters – Tools designed to shield malware and protect it from debugger analysis, static analysis tools, and emulator-based dynamic analysis systems.
- Binders – Tools that allow the malware author to embed the malicious payload in other distributable file formats (e.g., as an Adobe Acrobat file, Microsoft Excel file) and may include exploits to help with the installation of the payload.
- Evasion plug-ins – Tools, plug-ins and services that can be embedded or wrapped around the malware to ensure that it can evade signature detection and dynamic analysis system detection.
- QA – Third-party tools and cloud services that allow the malware author to dynamically test their malware against multiple anti-malware tools prior to use to ensure that no tools are yet capable of detecting the malware package.
- Suite development – The construction of attack suites that include the ability to construct unique malware, host command-and-control (C&C) services, manage attack campaigns, control multiple botnets, and coordinate data extraction of victims.
- Templates – Many malware suites include templates for orchestrating social engineering campaigns and drive-by-download fake sites (e.g., faking a bank's login screen). Third-party template packages are often available for regionalization and language support.
- Blacklists – Third-party supplied lists of IP addresses and domain names associated with security vendors and researchers to ensure that those addresses are never served malicious samples for their analysis.
- Exploits – Third-party subscription services to new exploits – both zero-day and recent high-profile vulnerabilities – that can be used to help install the malware on a victim's host, undetected.

Distribution services

Distribution services involve the process and practice of managing the distribution of crafted malware to potential victims.

- Fast-flux services – The provision of robust DNS services that help load balance and resist take-down efforts.
- Pay-per-install (PPI) – The business of purchasing access to website content (e.g., iFrames) to host malicious code or social engineering content designed to help infect a potential victim.
- Drive-by-download – The specialist business of malicious page hosting that often employs exploit code to cause the malware to be installed on a victim's computer as they browse the page or advertisement.
- CDN – Dedicated crimeware support content delivery networks (CDN) that specialize in regional distribution of malicious content.
- Anti-takedown services – A broad category of commercial online services designed to thwart security vendors and law enforcement from taking down malicious content and preventing attribution.
- Dynamic DNS – A mix of free and subscription DNS services that allow for rapid use and deployment of domain and host names – typically employed to bypass blacklists and takedown attempts.

Hosting services

Hosting services include the processes and provision of servers and services that allow the criminals to host their malware, C&C, data repositories, and other online elements of their attack or cybercrime business.

- Blackhat SEO – Just as in the legitimate marketing world, blackhat SEO service providers ensure that malicious pages and links are propagated to the foreground of search results and banner advertising campaigns.
- Proxies – Services designed to provide a degree of obfuscation or anonymity of important servers and content by using a series of network proxies to hide the final sources and destinations of the malicious content.
- Botnet C&C – The use of pre-existing botnets to provide network services such as fast-flux, proxying and VPN access to critical servers. This is effectively sub-leasing a section of a third-party's botnet.
- Hacked servers – Various hacking groups continuously look for vulnerable online services, compromise them and host third-party content on them for a small fee. These hacked servers are generally taken down rapidly, but while they are compromised and in operation, they often have good reputations and are rarely blacklisted.

Laundering services

Laundering services is the process of turning stolen data, credentials or financial information into something that can be freely and legitimately traded without attribution to the source.

- Data resellers – Once data has been stolen via malware, a myriad of data resellers can be used to monetize the stolen data. Factors such as data freshness, completeness of personal information, accompanying validation credentials (e.g., PIN and home address of a victim's stolen debit card), and how many times the data has previously been sold all affect the value. Payment for stolen data may take many different forms – only some of which are currency based.
  - Arbitration – Independent, trusted third-parties that act as brokers and arbitrate negotiations between the data thief and resellers. They often determine the value of the data and ensure payments are made between both parties for a fee.
- Card processing – Multiple cybercriminal gangs operate sophisticated credit and debit card processing rings. They purchase caches of stolen cards and use them to launder money, print clone cards and purchase physical goods.
  - Credit card validation – Banks and clearing houses are often quick to block or cancel stolen cards. Third-party service providers are used to test stolen credit cards to confirm whether they are still "live" and what credit limits may be present on the stolen card.
- Mules – Mules are often the unwitting work-from-home employees of organized crime. Their task is to help launder money and purchases. Multiple layers of mules in different countries are often used to thwart international law enforcement and protect the real criminals.
  - Check processing – Stolen or newly printed physical checks are sent to mules for deposit in to their accounts. With each deposit the mule then transfers a percentage of the deposited check to a different account – often another mule – for laundering money.
  - Package delivery – When cybercriminals obtain stolen bank cards or gain access to online retail accounts, they will often order high value items for delivery to a package mule's address. The package delivery mule then repackages or forwards that package on to another address.

A victim's view of the ecosystem

The commonly-perceived view of a malware infection is that the user is tricked into clicking on a malicious link that in turn takes their Web browser to a drive-by-download website. It then exploits a vulnerability in their Web browser to install a botnet agent that eventually steals all their personal data and uploads it to cybercriminals in some other country.

The above scenario, while all too often believed to be true, is a relatively naive and an increasingly outdated perspective on how the malware ecosystem operates to optimize its return on victims. This scenario does not describe the dynamics in which multiple criminal entities take control of their victims.

A more realistic scenario, and one that illustrates how many hands may be involved, is the following:

1. The user browses the Internet and visits a favorite website using a default Web browser.
2. A page on one of the favorite websites contains a banner advertisement. That banner advertisement was supplied by a third-party ad network that in turn is fed by an affiliate ad network, and one of those affiliates is operated by a criminal entity and serves up malicious ad content.
3. Upon rendering of the (good) Web page, the code of the (bad) banner advertisement is executed. The code silently exploits a number of vulnerabilities in the Web browser designed to:
   a. Download dropper code from an external website.
   b. Invisibly execute the dropper code to build the dropper agent.
   c. Silently run the newly-built dropper agent.
4. The dropper agent then performs the following operations:
   a. Disables the operating system's security update features.
   b. Disables the victim's anti-virus capabilities and prevents them from getting updates.
   c. Cycles through a pre-agreed list of download URLs.
   d. Downloads malware packages from any of the URLs that are active.
   e. Silently executes and allows the downloaded malware package(s) to install.
   f. Erases any Web browser and operating system log items that could indicate the installation of the dropper, vulnerability exploitation, disabled functionality, and the installation of the malware package(s).
   g. The dropper deletes itself from the victim's computer.
5. Each malware package that was installed on the victim's computer then:
   a. Inventories the victim's computer.
   b. Uploads the collected results to a botnet operator's C&C infrastructure.
   c. Downloads an update to the malware configuration file, containing new and additional C&C locations if one is present.
   d. Downloads an update to the malware agent if one is present.
   e. Retrieves any cached commands from the botnet operator and begins to execute them.
6. Periodically, each malware package will poll its C&C for new instructions and download and install any updates or new/secondary packages.

7. Access to the victim's computer will likely be leased, traded or sold to different criminal operators over time for reasons such as:
   a. A wide global net was originally cast, but the criminal operator may only be focused on victims in the United States, United Kingdom and other English-speaking countries – so they will sell or trade away unwanted systems.
   b. The operator has already extracted all saleable personal data on the victim's computer.
   c. The victim's computer has a high-speed Internet connection and is more valuable to DDoS and spam operations, or to serve as an infrastructure component for other botnet-related services.
   d. The computer is located within a corporate network and has access to multiple network resources, and is consequently more valuable to organized crime and state-sponsored syndicates.
8. Once all value has been extracted from the victim's computer, the system is deemed to be disposable and is used for noisy criminal operations that are likely to reveal the fact that the host is infected. Typical noisy operations include:
   a. Participation in DDoS and spam relay activities.
   b. Acting as a file repository or torrent node for pirated movie content, pornography and stolen commercial software (i.e. "warez").
   c. Providing anonymizing proxy services.

Throughout the lifetime of an infection, a victim's computer may be exposed from several days to multiple years of abuse and remote control by a wide variety of criminal operators.

During the lifetime of the infection it is probable that multiple malware packages will be installed on the victim's computer, and that network communications with a wide variety of criminal or malicious Internet hosts will be observed.

As illustrated by the previous figure, much of this activity can be categorized as pre-C&C monetization or post-C&C monetization.

Pre-C&C monetization

Pre-C&C monetization includes all malicious activities that eventually lead to successful communication between the malware agent and its C&C infrastructure. Key points to consider are below.

- Rogue ad networks will force the Web browser to visit a drive-by-download site. Purveyors of rogue ad networks typically get paid based upon the number of potential victims they route to the malicious destination. It is not uncommon for rogue ad providers to serve a single ad that in turn directs the unsuspecting victim to multiple drive-by-download sites operated by separate criminal organizations – thereby multiplying their financial return per victim.
- Drive-by-download site providers typically get paid for each dropper installed and may service multiple dropper providers simultaneously. As the victim's computer gets exploited by the malicious page, it typically receives instructions to download and receive more than 10 different payloads from different sites and criminal entities.
- Dropper site operators typically get paid for each software package or malware agent they successfully install upon a victim's machine. Therefore, it is common practice for independent dropper operators to simultaneously download and install multiple pieces of malware from various subscribers to their service.

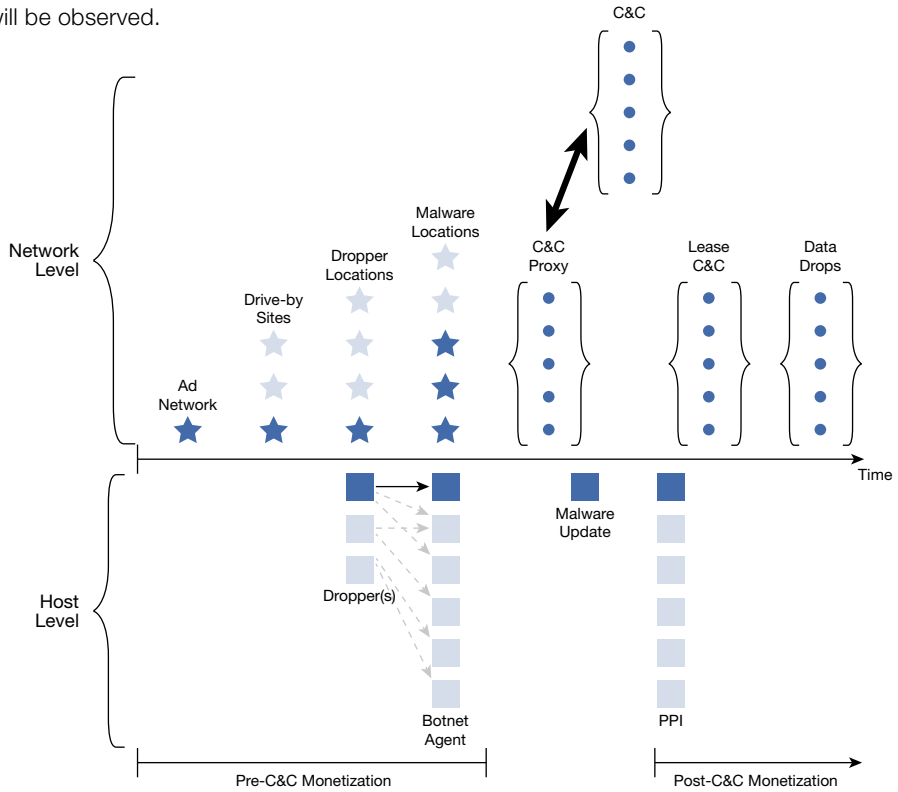

Figure 2: A lifetime of infection – pre- and post-C&C monetization.

Post-C&C monetization

Post-C&C monetization includes all malicious activities that follow, from a botnet agent being installed to successfully communicating with the malware's C&C infrastructure. Key points to consider are below.

- Most botnet operators update their installed malicious agents on a regular basis as a means of avoiding detection by anti-virus tools and to increase the functionality and features of the agent. As part of the process, the botnet operator may also update configuration files – usually encrypted – that contain the list of C&C servers the agent should use in the future.
- Botnet agents rarely connect to or communicate directly with the core C&C infrastructure of the criminal. Instead, C&C communication will be passed through a number of – and potentially multiple layers of – disposable proxies. These proxies are typically other botnet victims and hacked servers – thereby making it more difficult for security researchers and law enforcement to perform takedowns or provide attribution.
- Once a victim's computer has had all valuable and resalable information on it stolen, many botnet operators choose to install other criminal's software on the host. In this permutation of the PPI business, the botnet operator receives funds for each package they install. While many of the packages may be malware agents from other criminal operators, sometimes they may install other grey-ware software to earn credits in online games.
- At the network level, the current botnet agent, or other agents the primary operator has chosen to install, may provide leased access to the victim's computer to other criminal organizations. In many cases, these third-parties may be given access to the host via a separate and distinct C&C infrastructure, or they will install their own botnet agent that uses a different C&C infrastructure.
- At various points within the malware lifecycle, the malicious agent will require the host to transfer files and other data to a remote site. This upload activity may be related to the theft of personal information, data extracted from corporate databases and file servers, the results from lateral network scanning, or logs of other nefarious endeavors. The botnet operator typically operates a separate network infrastructure for these activities that are not associated with their C&C infrastructure.

As a consequence of these pre- and post-C&C monetization opportunities, the victim's machine will transfer files and receive C&C communications from multiple Internet servers throughout the infection lifecycle. There is rarely a one-to-one correlation between an initial infection vector and a single botnet agent.

Throughout the infection lifecycle the victim's machine will also receive many different malicious agent updates – often from multiple criminal entities. Because of this, it is increasingly difficult to pair observed network traffic to a particular malware agent at a given point in time.

Malware sample acquisition

Practically all commercial anti-malware defense development is predicated upon having access to a stream of malware samples recovered from the field, such as through customer submission, honeypots, network interception, milking sites, monitoring underground forums.

As such, the timely and efficient capture of malware samples is a critical component in a vendor's ability to provide protection to their customers.

Over the last four decades, information security vendors – primarily anti-virus vendors for the first two decades – have developed their own ecosystem for acquiring and sharing malware samples and information about their infection vectors.

Below are some of the most common methods of new malware sample acquisition.

- Customers of host-based and network-based anti-virus products automatically upload suspicious files and samples of captured malware samples to their vendor whenever a signature is fired. These triggers are typically based on file heuristics, packer identification, behavioral characteristics, download source reputation, and non-whitelist inclusion.
- Customers of email and Web browser protection suites constantly evaluate the network reputation of URLs and Internet resources. Each URL is assigned a unique hash that is checked against a cloud-based whitelist of hashes. If the hash is known to be bad, the client software blocks the URL. If the hash is unknown, the client software sends the URL to the vendor for cloud. The vendor then downloads any files associated with the URI via their cloud and independently assesses it for malicious intent, thereby harvesting new samples.
- Security vendors operate honeypots, honey-accounts, spam bots, and other passive services on the Internet. Appearing as attractive targets to multiple criminal operators in the malware ecosystem, they wait to be attacked, enabling the capture of new malware samples.
- Security vendors run dynamic systems such as honey-monkey crawlers, watering hole spies, and updater milking operations that continually probe known sources of badness and malware release points for new sample acquisition.
- Cloud submission services allow anyone to upload a suspicious file and have it automatically checked and scanned by multiple anti-malware vendor products for maliciousness and threat classification. Submitted files are then shared with the vendors that contributed detection tools to the cloud service.

Technical note

Any commercial deployments of anti-malware products specify contractual limitations that prohibit suspicious files or malware samples that may be captured within their network from leaving their network (e.g., samples can't be shared with their vendor). These contractual limitations may also specify that the vendor cannot share those malware samples with any other member of the vendor ecosystem.

New and unique malware samples captured using the above techniques and many more are then made available for sharing between security vendors. Such sharing agreements initially came about in the 1980s as an informal arrangement between the then-largest half-dozen anti-virus vendors, but have since blossomed into a considerably more diverse sharing ecosystem – including free and commercial paid-for subscription services.

Malware sample freshness

In the perpetual battle between security vendors and the malware ecosystem operating around the world, the dynamic battlefield means that there are many external events that can have an adverse effect on the operation and value of a particular malware sample.

Just as innovation in malware armoring techniques have countered advances in static analysis and automated behavioral processing of malware, information-sharing and standardized takedown agreements have enabled security vendors and law enforcement to quickly cripple and contain certain types of malware outbreaks.

As such, from an anti-malware research perspective, the freshness of malware samples – both the harvesting and subsequent processing of samples – is a key element in the development of defense updates.

For security researchers and vendors to develop detection updates or craft new anti-malware defenses, access to the broadest range and timeliest samples for analysis is a constant concern.

The longer the malware has been out in the field, it will often correlate to a higher number of victims and it will become more likely that captured malware samples will be broken when processed for research purposes – thereby making them less valuable.

Since malware sharing between security vendors plays such a critical part in sample acquisition (e.g., global diversity, variant capture, regional targeting), it is important to understand what phases of capture and sharing affect the freshness of samples.

Overall, there are a number of transition phases in the deployment and handling of malware that introduce delays in the timeline and affect the subsequent value of the sample. These include:

Delta between when a malware was created and released

Cybercrime gangs may bulk-produce and QA malware prior to release in a campaign. Some malware samples could have been created weeks in advance of use against their targets. These samples may have been undetectable at creation, but because other variants may have been released in the intervening time, detection updates may have also been released.

Delta between malware kit and malware variant

Malware authors may offer construction kits for their malware for sale and distribution. These construction kits are designed to produce custom variants of their malware.

Delta between released and first infected

Depending upon the operator's efficacy of deployment (e.g. failed SEO campaign, poor drive-by-download site construction) a malware may be released for a period of time before it successfully compromises a victim. Alternatively, the malware may in fact be broken and never successfully infect a victim.

Delta between first infection and widespread infection

Some malware strains, particularly new families of malware, may be released by their creators to target specific environments. Once the malware has successfully proven its efficacy, the creators may broaden the scope of systems and delivery channels they use.

Delta between infection and host detection

Because malware sample harvesting is often initiated through some partial or suspicious behavior detection, there is often a period of time between infection and when a malware sample – a variant or family – may actually be detected and classified as a threat by the infected host.

Delta between host detection and vendor sample submission

Factors such as network connectivity, malware file size, and local host manipulations of the malware itself may introduce sizable delays between the malware being detected and before it can be safely submitted to a vendor for investigation.

Delta between sample submission and private sharing between vendors

While a malware sample may be automatically shared with a vendor, it may take some time before that sample is shared as part of any private sharing agreements between vendors.

More often than not, in the case of potentially new malware families, the initial vendor will schedule the sample for analysis and, if determined to be a new family or potentially newsworthy, that vendor will develop new detection and classification updates prior to sharing the sample with other vendors.

Delta between sample submission and signature development

It takes time to determine whether a new malware sample is new and worthy of additional signature development. Once that determination is made, the malware is scheduled into a cycle for feature analysis, signature development, and subsequent detection validation – before the signature is provided as an update to customers.

Delta between signature development and update availability or installation

A new signature or detection engine must be tested for efficacy against the threat, and verified to not cause false positives. The process may also require tuning or replacement of existing or outdated detection signatures and models. This process consumes time, and it may be several days before an update is available for the general installed base.

Delta between private and public sharing

Sharing with private partners is predicated upon whether the original vendor has signatures to detect the threat. Once that detection is available, the sample may be shared via private-sharing agreements.

Public sharing typically only occurs some period after it has been shared privately – generally to ensure that all partners and affiliates of the private-sharing collective have developed their own detections or protection against the particular malware threat.

Exceptions may be made for notable new malware threats (e.g. a new malware family) that are likely to be of marketing and public relations value – in which the vendor's researchers provide a detailed analysis of their findings for public education. Even then, malware samples of that particular threat may be held back several weeks and only available to trusted anti-malware vendors.

Delta between public sharing and public analysis

Having publicly shared a malware sample – usually through some commercial feed agreement – it may be a period of time before security researchers identify and note interesting features of the malware sample (e.g. a notable evolution of a threat, such as the addition of MBR encryption) and that analysis is made public.

Technical note

Some vendors, while limited from sharing a captured malware sample or disclosing the source of their analysis, may share non-attributable information such as the sample's hash value with other vendor's analysts.

In this way, third-party vendors may encounter samples in their own customer networks and correlate threat attributes with each other. Such a mechanism is used extensively for tracking advanced persistent threats (APTs) and malware that appears to target critical infrastructure – without divulging who the targets are.

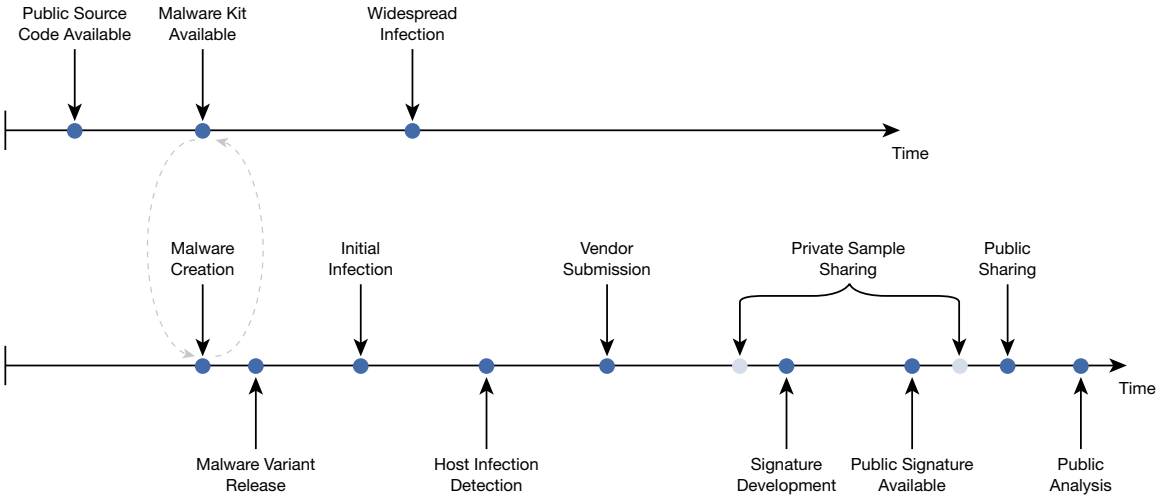

Figure 3: The respective timelines and typical sequencing of malware sample handling and sharing.

Dealing with broken malware

While many millions of malware samples are captured daily, only a few million per day are typically executable – meaning that they contain code and can be installed upon a target's device in such a way that they are capable of performing their malicious tasks.

Suspicious files that are likely to be malware – as determined by the reputation of the download source URI or when downloaded by a known malware instance – and are not executable in their own right are often labeled *broken*.

File heuristics and static analysis techniques can sometimes be used to process these broken malware samples to determine some of their malicious characteristics. However, more detailed analysis – including dynamic and behavioral analysis – is not possible.

A lot of broken malware samples are acquired through network-based sample collection processes – in particular, network packet capture analysis and extraction, and the milking of known malware distribution sites. Common reasons for broken malware include:

- Malware update files may be encrypted and must be decrypted by the currently installed malware on the victim's host before it can be replaced.
- The downloaded file may be a new configuration file or malware library (e.g., a DLL) that is encrypted.
- The file download was interrupted for some reason, so the file is only a partial capture of the malware.
- The downloaded file may be locked to a target computer – using similar principles to commercial software licensing practices and TPM locking.
- The intercepted file was only one of multiple components that needed to be downloaded in order for the malware to run. Since other components that may have also been downloaded were not collected and distributed together as a single malware sample, the context of each sub-component is lost and they cannot execute independently.

Since broken malware cannot be executed and run, it is impractical to extract network behaviors and characteristics of the threat.

Malware network feature extraction

The extraction of network features from captured malware samples is typically achieved through static or dynamic analysis, assuming that the malware under study actually has some kind of network functionality, and not all malware does.

In general, standard malware armoring tactics and the way in which network configuration information may be stored within a packed executable file make dynamic analysis the preferred and most illuminating method of analysis.

Dynamic analysis of malware is performed via virtual machines or bare metal machines – and the entire analysis process is often referred to as *sandboxing*.

These analysis systems are typically installed with the most common operating systems that attackers target, along with popularly exploited software packages, such as Adobe Acrobat, Microsoft Office, and Sun Java.

Each analysis system is also instrumented in such a way as to allow malware samples to be executed freely and all host-based activities are recorded for analysis afterwards. Network-based instrumentation is used to monitor outbound or lateral network behaviors and are typically stored in PCAP format for later automated dissection and classification.

As sandbox analysis became more prevalent in vendor analyst and enterprise network environments, malware authors and evasion tool developers have invested greater efforts in their evasion. Some common evasions include:

- Recognizing that the malware is operating within a sandbox and playing dead or operating in a benign way. Sandboxes can be recognized by many factors, including video drivers, debugger presence, monitoring instrumentation, license keys, video screen sizes, browser history, and registry dates.
- Comparing Internet availability and access to ecosystem blacklists. If outbound network traffic is filtered or the source IP is associated with an anti-malware or security research lab, the malware refuses to work.
- Some malware may target a specific region. Executing the malware from any other geographic location, or having unexpected regional configurations (e.g., keyboard settings, language settings), results in benign activity.
- Targeted malware – including malware updates – is often locked to a specific machine or machine configuration. Some locking systems are as sophisticated as the licensing systems of popular commercial software.
- Waiting for some period of dynamic, usually user-based, activity before activating and communicating externally. For example, waiting for the first reboot of the month, accumulating 1,000 key presses, waiting for the 10th email by the user to be sent.

Malware samples are executed in these dynamic environments at a point in time – meaning that the observed behaviors of the malware at the host and network level are only applicable to the time of execution.

The dynamics of the malware support ecosystem, the infection lifecycle, and the malware sample acquisition process all combine to influence the output, or lack of output, from any studied malware sample.

Assuming the malware sample is not broken, is not sandbox-aware, and successfully executes in such an environment, the network features of most interest to security analysts and detection providers include:

- Primary artifacts – Destination domain names and IP addresses.
- Secondary artifacts – Ports, protocols, and unique strings (e.g., browser user agents, URIs).
- Tertiary artifacts – Traffic samples, typically in PCAP format, used to test detection efficacy and the development of machine learning models.

Malware network feature confusion

The network artifacts extracted from a malware sample are often subject to interpretation because there are multiple execution and external events that can have a dramatic effect on the analysis and usefulness of the data.

Some of the most common execution paths and external events that adversely affect the quality and interpretation of network sandboxing results include:

- The malware under examination is simply a *dropper*. As part of a drive-by-download, or similar infection vector, the malicious payload is a dropper file whose purpose is to install a small executable package – typically a downloader – on the victim's machine.

  Without the continued execution of the original drive-by-download attack script, once the dropper has installed its package, there is no further running of the payload and hence no network activity. Droppers are often confused with broken malware.

- The malware under examination is a *downloader*. As part of an infection vector, or as part of an update to an infected host, the malicious payload is built to reach out to a specific URL typically over HTTP or HTTPS, download the file and execute it on the victim's host. The URL observed through the sandbox analysis is referred to as a *malicious download URL*, and is sometimes confused with a C&C location.

The post-download traffic, associated with the newly installed malware agent that is oftentimes a botnet agent, may be an aggregate of traffic from both the downloader and the botnet agent operating on the host. Confusion exists when these network behaviors are associated with the original downloader malware rather than the downloaded package.

- The malware package may initially try to reach its malicious download URL or botnet C&C by attempting to resolve a series of domain name requests. Depending upon the age of the malware, some or all of the domain names relied upon by the malware may have been taken down by anti-malware researchers and law enforcement. As such, the domain names will never resolve, resulting in DNS-only traffic (e.g. NX = no such domain).

  In an increasing number of instances, these taken-down domain names may be sinkholed by researchers and law enforcement in order to study the botnet. *Sinkholed* domains are successfully resolved by DNS and direct the botnet agents to IP addresses capable of answering some, if not all, requests by the botnet malware. Confusion exists when sinkholed IP addresses are interpreted as live malicious servers owned and operated by criminal entities.

- The cybercriminals behind the malware may have relied upon or utilized part of the ecosystem that leverages hacked servers and other infected victims to host the C&C, malware updates or new configuration files. Hacked servers are usually only a temporary repository for an attack as they are assumed to be fixed or remediated within a few hours or days at most.

  They are used because they commonly provide high bandwidth and are typically trusted sites that are often whitelisted by security companies. Confusion arises when the malware is communicating with a formerly hacked site or infected IP address, and are misclassified as belonging to cybercriminals.

- Malware updating during the sandbox analysis can cause much confusion, especially when multiple malware agents are downloaded and executed within the same session by the initial malware sample. If executed within the same analysis session, it is inevitable that primary and secondary attributes will be associated with the wrong malware sample.

Network-based detection and classification of malware artifacts

The path in which malware transitions from a creation and distribution ecosystem, through sample interception and sharing, and on to dynamic analysis, is a complex and time-sensitive journey.

Each phase of this journey has an impact on how artifacts extracted from the malware may be used as well as their usefulness. It is unlikely that two malware samples are guaranteed to follow the same path.

From a network-based malware detection and prevention perspective, it is critical that all these journey factors are considered and accounted for – as they tend to have a strong influence upon the validity of subsequent threat classification and response of any findings.

Domain-to-IP correlation

Correlation between suspicious network events and a known malware threat can be difficult; timing is everything. Given the dynamics of the malware ecosystem and the inherent problems of sample freshness through the malware-sharing schemes that drive feature extraction systems, timing plays the most critical role in understanding and responding to a malware-borne threat.

Given a single malware sample, ecosystem factors that influence the value and usefulness of primary network artifacts (i.e. domain names and IP addresses) derived through dynamic analysis include:

- Malware configuration files are updated frequently with new downloader, uploader, and C&C sites in response to factors such as takedowns, system patching, dynamics in hosting services, and proxy services. As a result, domain names may become redundant for malware operation or may be recycled for new malware campaigns (both delivery and C&C) at any time.
- Botnets may be subdivided, resold and leased to different operators who will use their own preferred C&C services. While the malware component has likely remained the same, the victim's host is now under some other criminal's operation and may be clustered and managed as part of different botnet.
- Domain names may be used by malware before they are live or registered. After the malware is distributed and believed to infect a large number of hosts, criminals will start up the C&C services, update DNS and, for a period of time, direct malware traffic to these temporary servers.
- As IP and DNS reputation systems become more popular, criminals may register domain names well in advance of malware ecosystem use and point them at well-known whitelisted sites in the hope that their new domain names will inherit some of their good reputation. At some point in the future, they will modify DNS settings to point to their real C&C server and deployed malware will connect and receive new instructions from the C&C.
- The malware sample may have been shipped with a list of rogue domain names that are associated with less popular, but real websites, for the purpose of tricking security vendors to blacklist legitimate sites (e.g., targets for fraud).
- Some malware families include domain generation algorithm (DGA) techniques to automatically generate time-based C&C candidate domain names. On a daily basis, the malware probes the new list of candidate C&Cs with the expectation that criminals will activate a C&C server to issue new commands. Multiple one-use domain names are created each day, making it impractical for security vendors to operate blacklists.

Of course, if the victim's host is compromised there are probably multiple pieces of malware installed – all exhibiting their own overlapping C&C dynamics.

In Figure 4, the first time the domain name will be looked up (a.ip using A.ns as its authoritative name sever) will be by the malware author to test DNS settings. Twice through this example timeline, cybercriminals will point their domain name at popular whitelisted sites (b.ip and c.ip) to inherit their reputation.

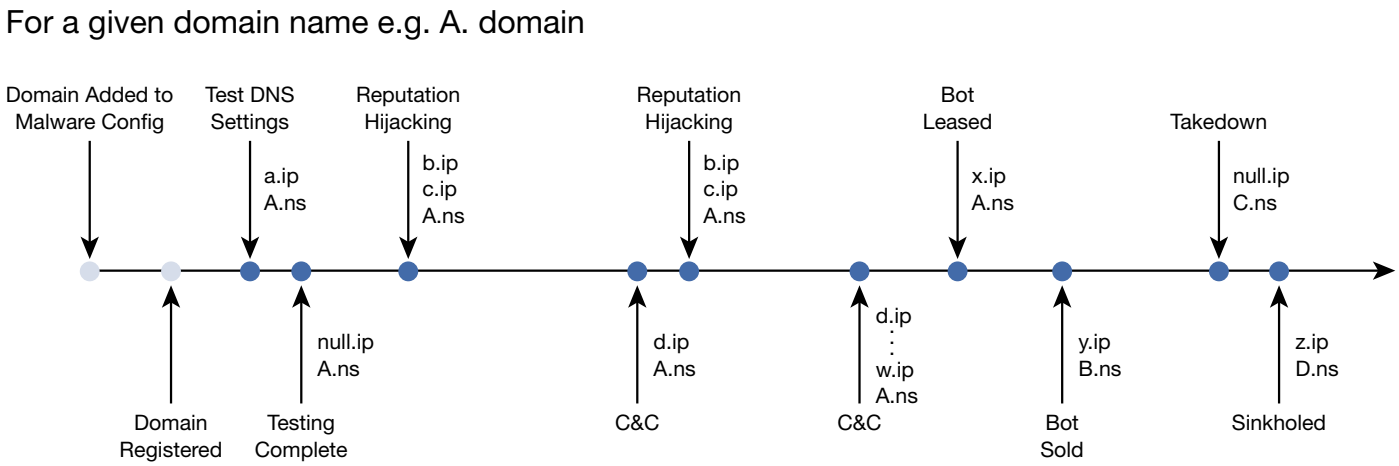

Figure 4: For a given domain name, there are multiple times in the malware lifecycle that it will point to different IP addresses (a.ip through to z.ip).

When the botnet is in full operation, multiple IPs are used to host C&C functionality (d.ip through w.ip). At some point the botnet is leased to a third party – noting that the C&C IP changes (x.ip) but the domain is still controlled by the same name server (A.ns).

Later, the botnet is sold to another criminal operator whereupon the C&C IP and name server both change (y.ip and B.ns). Toward the end of the lifecycle, the botnet C&C is taken down by a security vendor (null.ip and C.ns), before finally being sinkholed (z.ip with a new name server of D.ns).

Figure 4 also shows that the domain name associated with just one C&C channel of a sample botnet agent can have multiple IP addresses and name servers throughout the malware's lifecycle.

In general, the longer a malware sample has been installed and the more valuable the victim is (e.g., corporate asset with access to sensitive and easily sellable data), the more times its C&C information will change – as the compromised device is passed between various criminal operators.

In addition, the larger the botnet and the greater its global distribution of victims, the more attractive it is to threat researchers and vendors. Access to sinkholed data from large botnets is keenly sought after by various commercial entities – so sinkholed domains associated with these large botnets are often managed and traded between various threat intelligence service providers.

Criminal operators of fast growing botnets often divide their populations of infected victims frequently for management (e.g., smaller botnets are more responsive and easier to operate) and evasion purposes (e.g., threat researchers tend to focus on "big" botnets). As such, with each division of the botnet, different C&C servers and infrastructure may be introduced into the equation.

The effects of timing

Ignoring most of the dynamics of the malware ecosystem for the time being, from the perspective of an analyst charged with investigating a malware, there are effectively three independently operating timelines that affect analysis.

These three independently shifting timelines are:

1. The malware's C&C domain name and IP synchronization timeline (see Figure 4).
2. The third-party malware sample capture and sharing timeline (see Figure 3).
3. The victim's timeline covering asset infection, discovery and eventual analysis (see Figure 5).

The only timeline the threat investigator has any direct control over is the post-discovery period of the victim's timeline. Depending upon the time the analyst begins the investigation, the results can differ significantly for the threat.

In many malware outbreaks where the victim is patient zero, the victim's timeline of discovery may in fact be the start of the third-party malware sample capture and sharing timeline.

In Figure 5, timelines for C&C domain name and IP usage in the malware, malware sample handling and sharing, and the victim's host infection timeline overlap at various points in time. Depending upon when an analyst is investigating a post-discovery malware infection and looking up intelligence on the threat (e.g., malware sample collections and C&C details), they will receive different results.

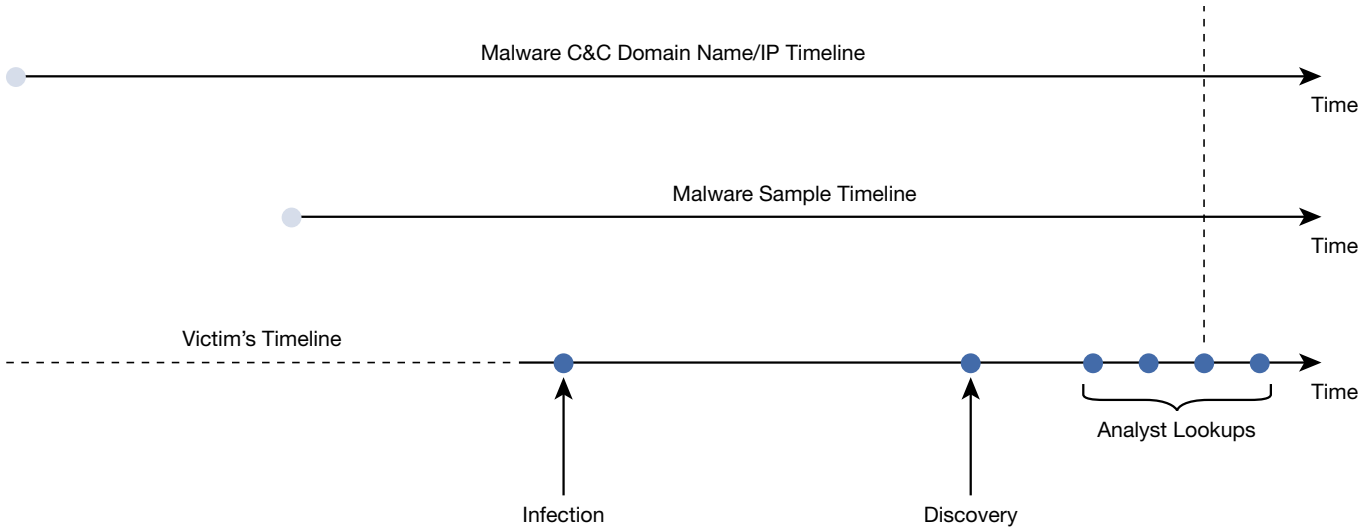

Figure 5: Overlapping timelines.

It is important for the analyst to identify how long a victim device has been infected, but the calculation of risk – and the requirement to disclose breach details – are based upon what the malware threat was capable of performing preceding the discovery date. For example, was the C&C for the threat already sinkholed or was it in the hands of a particular criminal organization?

While an analyst may be able to search through corporate historical DNS logs and mine packet captures for relevant details, passive DNS data (pDNS) provides a historical timeline of domain name-to-IP associations with a global perspective. This pDNS perspective on the malware's C&C can be used for threat classification and attribution to criminal operators scattered throughout the malware ecosystem.

Armed with a list of IP addresses, as shown in Figure 6, it is possible to iterate through pDNS to identify additional domain names associated with a cluster of IP addresses (B.domain to E.domain), and from there identify more IP addresses likely associated with the threat or criminal operator.

Detailed forensic analysis of the victim's host will typically yield malware samples. Armed with these samples, the analyst can easily verify vendor anti-malware coverage.

Using the unique malware hash (e.g., MD5, SHA1), an analyst or queryable cloud service can be used to enumerate additional C&C domains associated with the sample and, from there, query to identify other known global malware samples that have been observed to use the same C&C domain names or IP addresses.

Figure 7 shows that it is possible to iterate through the process again to identify related domain names (B.domain to G.domain) associated with a collection of related malware, and so on, in order to better understand the tools and tactics of the criminal operators behind a single captured threat.

Typically, the link between a captured malware sample and C&C domain names – and other associated third-party malware samples – is not a one-to-one relationship.

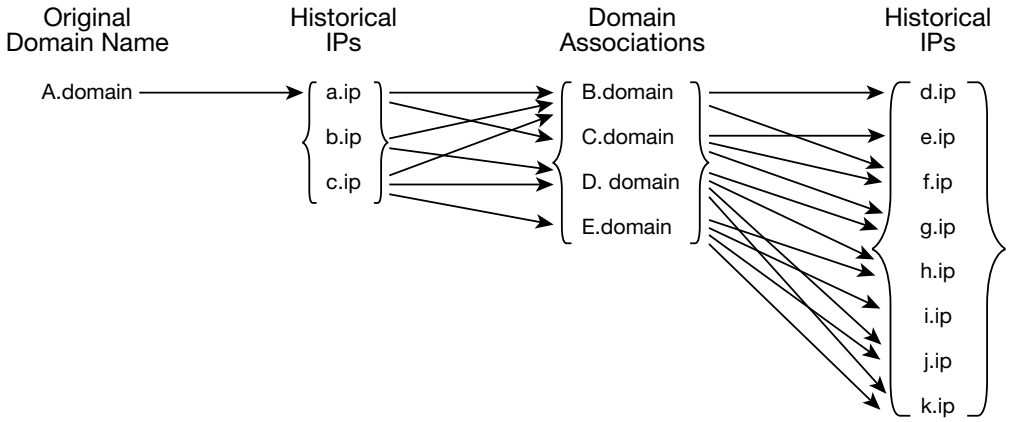

Figure 6: Use of pDNS databases to enumerate historical IPs (a.ip, b.ip, c.ip) associated with a domain name (A.domain).

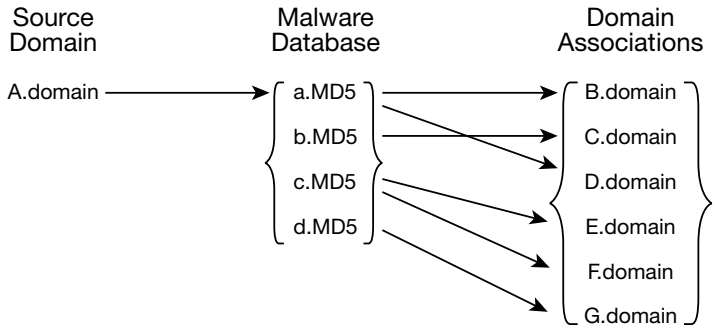

Figure 7: Use of cloud malware databases can yield additional malware (e.g., a.MD5 to d.MD5) associated with a particular C&C domain name (A.domain).

Things become more difficult again as any of the previously analyzed malware samples behaved deceptively in their automated analysis (i.e., the malware sample(s) rigged their domain names and IP address results to throw off analysts).

Technical note

Note 1

Host-based forensics process often misses the downloader, dropper, and associated infection vector log details due to the malware's installation clean-up processes.

Note 2

Cloud and vendor processing of malware samples, particularly dynamic analysis, is typically performed once and at a fixed point in time. Therefore, the effects of the malware C&C domain name and IP timeline also play a significant factor in correlating an analyzed malware sample (with its extracted primary network features), and using those primary features as a correlation point with other malware samples. This is one reason why analysts iterate through multiple levels of associations between pDNS IP addresses and domain names, and multiple levels of MD5 hash-to-domain name associations in order to understand the nature of the threat.

The effects of GeoIP

A lot of public attention is placed upon the geographic source of a malware threat. Numerous IP-to-physical geographical location (GeoIP) services exist to assist with this correlation.

GeoIP services are relied upon extensively for online advertising – typically for ad localization services. Since IP addresses have not generally been allocated in geographic batches, GeoIP service providers have invested extensive resources into IP enumeration tools and feedback systems to help correlate between an IP address and a geographic coordinate.

As such, GeoIP is usually precise enough for marketing campaigns targeted at a city level – based on residential and subscriber network locations. However, GeoIP services are much less precise (i.e., country or continent level precision) for Internet infrastructure and hosting facility locations, and may be substantially less precise when dealing with IP ranges associated with global companies (i.e., registered headquarters country details).

Common reasons why GeoIP precision fails outside of residential network identification include:

- The allocation of IPv4 addresses is exhausted. Internet corporations are constantly buying and selling ranges of IP addresses around the world to meet their hosting needs. Since no central authority exists for logging such sales nor tracking the geographic allocation of these sold or traded IP addresses, there is no static point of reference.
- Groups and ranges of public IP addresses are typically managed through Autonomous System Numbers (ASN) by one or more network operators. Exterior routing protocols are used to exchange routing information between Autonomous Systems (AS). While AS and ASNs are labeled – including the owner's name and address details – the geographic information in the address details is typically the company's main business office and not the location of a host facility or facilities, which may be somewhere else in the world.
- The growth of cloud computing and service provisioning means that global providers are using pools of ASNs and dynamically segmenting networks and allocating IP addresses to physical servers around the world. Within their collective ASNs, an IP address may be dynamically allocated to a new host at any point in time.
- Privacy-preserving services have become more popular. Grey market and black market anonymizing services – including bullet-proof providers – frequently cycle through ASNs and never reveal their host service locations.
- IP addresses that have no precise geographical location history (beyond routing information to a particular country's national telecommunications infrastructure) are typically set to the country's geographic central coordinates (i.e., the location deemed to be the center of the country or city).

  This has also produced numerous attribution and legal problems for those residences that unfortunately happen to be closest to these arbitrarily assigned center points.

- Distributed communities assigned a pool of IP addresses may have their entire allocation of IP addresses assigned to a single commercial or residential address within that community – due to factors such as being closer to a major highway and having an open Wi-Fi that was included in a GeoIP mapping exercise at some point in time.

  These communities, which may encompass several thousand IP addresses and be used by a community spread over several hundred to several thousand square miles, are poorly represented in GeoIP databases.

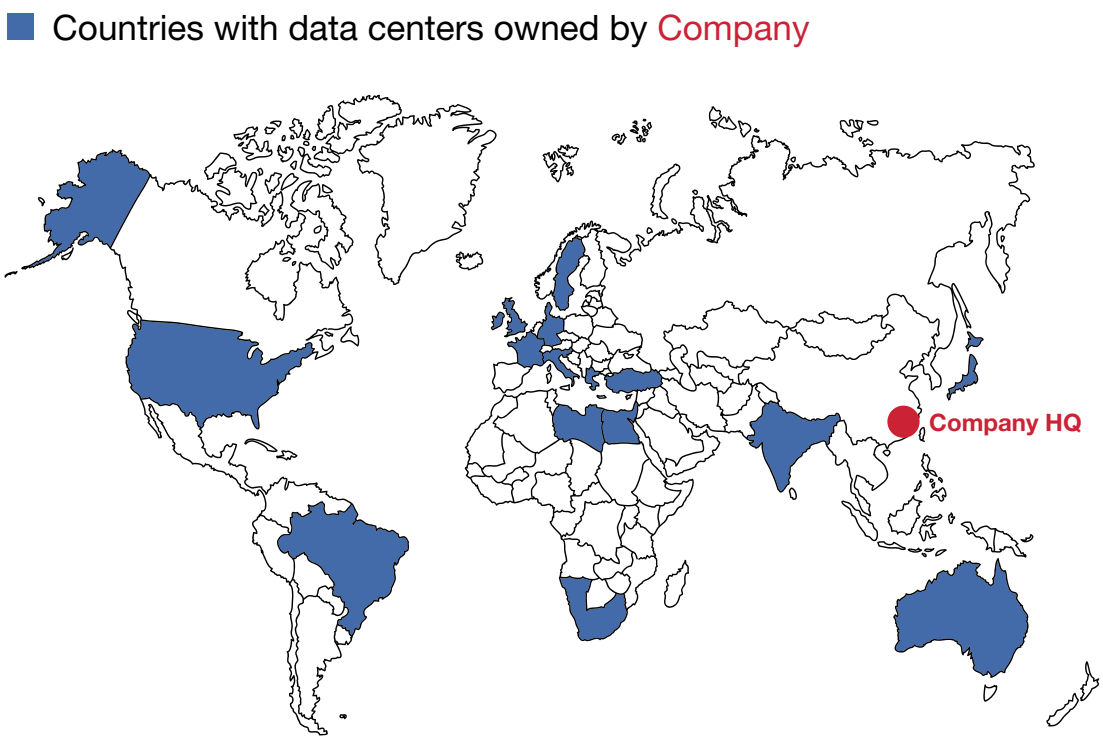

Figure 8: A multi-national Chinese company owns several Class-B IP address ranges and assigns them to the data centers they operate around the globe.

Figure 8 shows that each ASN is marked as owned by the Chinese company with its Beijing headquarters address. Based upon provisioning requirements and load, the company regularly reviews and reassigns IP addresses between data centers. All the IP addresses are misidentified by GeoIP tools as being located in Beijing.

From a threat detection and analyst perspective, GeoIP also suffers from a number of problems including:

- Since GeoIP is based on an IP address, the time in which the IP address or domain name associated with an IP address is observed in use by a malware sample also suffers from the same timeline reconstruction restrictions. Any associations between a malware sample and a geographic location must be defined in terms of the period in which it was observed.
- The malware ecosystem makes extensive use of commercial hosting facilities around the world, ranging from bullet-proof hosting to hacked Internet servers, to distributing malware and controlling its victims. These hosting arrangements operate to deliberately confuse any location-based services and thwart law enforcement efforts to takedown or prosecute them.
- Criminal operators behind botnets and widespread malware threats make extensive use of proxy networks to anonymize traffic and protect core C&C servers from being attacked or taken down.

Network communications between a botnet victim and its authoritative C&C may pass through multiple hops, only the first of which (i.e., closest to the victim) will be visible to the analyst. This first hop is inevitably not the real C&C server, and so its location is largely irrelevant to understanding the threat.

- Malware that is geographically sensitive (e.g., checks the language of the victim's computer) is also targeted geographically. As such, different behaviors and C&C infrastructure may exist in different parts of the world. The same malware may use different C&C IP addresses when executed on machines in different countries – making GeoIP a difficult feature for correlation.
- Security companies and law enforcement have become much more efficient in taking down infrastructure of the malware ecosystem. The use of honeypots and sinkholes has grown considerably. If the malware sample has had its C&C infrastructure sinkholed, the GeoIP data will be representative of the sinkhole's geographic location and not the location of the original C&C threat.

The GeoIP relationship to domain name and then to a malware sample is tenuous, and without using time-synced DNS records (e.g., when was the malware actually constructed and pointing live to an address, versus when was the malware analyzed in a sandbox for behaviors) it cannot be used for meaningful analytic connections. At best, it is a circumstantial observation that provides a history of possibly related events when there is no stronger feature available.

IP network artifacts of value

While GeoIP may not provide a meaningful attribute to the threat classification and response of malware intrusion, there are a number of other IP-related data that can be used to better understand the nature of the threat. These features include:

- The IP address is part of a residential subscriber network, which is the most accurate data that GeoIP services can provide. Use of such a residential IP address is likely indicative of at least one of three things: The remote address is a similarly compromised host, the IP address is a C&C proxy, or the attacker is an amateur who doesn't know how to obscure the source IP address.
- The IP address is part of a content delivery network (CDN) or cloud-provider network. Malware-derived traffic and network features pointing to CDNs likely mean that any level of attribution will be impossible, and potential blocking of the threat by IP address would be impractical.
- The IP address is associated with a security company, research network or sinkhole infrastructure. This means that the observed threat was, at some point in the past, a real threat, but has since been mitigated by a third party.
- The IP address is part of a known proxy, relay or anonymous exit node network. Outbound communication to these IP addresses are indicative of a professional malware operation.
- The IP address is within a physical distance of the customer network (e.g., a Starbucks 15 miles from the office). Such a discovery allows the analyst to quickly classify the threat as *internal* and can often be rapidly investigated and remediated. As such, the event is often highly actionable.
- The IP address is listed on several common or popular commercial blacklists.

Network-based threat attribution

When it comes to spotting malware-oriented network threats, the features extracted from previously caught and classified malware are critical components to detection. Depending upon the type of network monitoring technology being used, the primary, secondary or tertiary features of the threat's communications may be leveraged in identifying an attack in progress or an aspect of its C&C.

As automated malware analysis tools have evolved and become better at feature extraction, threat detection systems have similarly evolved to make better use of those features – as a way of increasing detection efficacy, labeling and attribution.

Network-based threat detection technologies make extensive use of signature-based detection systems – where these signatures are based upon and tested against previously observed and extracted malware features.

While there has been a tendency for vendors to distance their product technologies from the word "signature," they continue to use various classes of signature detection techniques.

Technical note

All signature-based detection systems share the need for experienced analysts to correctly group and classify malware samples prior to developing a signature. Failure to do so leads to future false positive and false negative detections.

There is also a shared assumption that analysts who perform the triage will also correctly label the threat. This ensures that a subsequent signature detection can be associated with a specific threat or category of abuse, or attributed to a specific entity.

To understand how network-based threat detection and attribution works, it is important to understand how the various approaches to signature-based detection have evolved and what their strengths are.

One-dimensional signatures

Blacklists and whitelists are examples of one-dimensional signature systems. They are found throughout security and exist in practically all detection and protection technologies.

One-dimensional signatures offer the fastest and most efficient way to categorize a data artifact (e.g. a domain name, IP address, user-agent, or MD5 hashes). As a Boolean operation, what you're looking for is either on the list or it is not.

Two-dimensional signatures

Classic regular-expression functions and string matching are examples of two-dimensional signature systems. They are the fundamental building blocks of anti-malware, intrusion detection, and data-leakage detection systems.

In malware, they are often used to search a binary file for known strings that help to label the type of threat it represents. Two-dimensional signatures came to the fore as a means to detect network-based threats within the content-level of traffic – easily capable of identifying previously known exploits and host enumeration techniques.

Data leakage prevention (DLP) is a more recent security technology that relies heavily upon two-dimensional signatures. Messages and file attachments are often scanned for specific strings (e.g. serial numbers or passwords) or construction formats (e.g. social security numbers of the format nnn-nn-nnnn with a regular expression of ^\d{3}-\d{2}-\d{4}$).

Multi-dimensional signatures

Security vendors developed a hybrid system as the threat spectrum grew and attackers found new ways to obfuscate the elements of their attacks that were most exposed to one-dimensional and two-dimensional signatures.

Instead of triggering on a single signature, a multi-dimensional signature was created. In sandboxing and network behavioral monitoring, certain actions and activities are labeled as either suspicious or bad.

When a threshold of good or bad activities is reached, the threat is classified and labeled. For example, a suspicious file is executed within a virtual environment. The file attempts to write to the Windows registry (neither good nor bad), add a file to the Windows startup path (suspicious), disable Windows updates (bad), read from the user's contacts list (neither good nor bad), and then send email to every address listed in the contacts list (bad).

Together, all of these individual actions (i.e. signatures) are combined and tallied and a decision is made that the suspicious file is in fact malicious and most likely a spambot.

n-dimensional signatures

Instead of manually trying to figure out and label all the good, bad and suspicious behaviors, a machine is fed an assortment of *known bad* and *known good* samples, which could range from binary files and network traffic to photographs.

The process then takes and compares all the observable behaviors of the collected samples, automatically determines which behaviors were more prevalent or less prevalent to each class of samples, calculates a weighting factor for each behavior, and combines all that intelligence into a single model of *n*-dimensions – where *n* is a variable size based upon the type and number of samples and behaviors the machine used, as well as overall complexity.

Different sample volumes and differing samples supplied over time will often affect n. In machine learning terminology, this process is called *supervised learning*.

	Primary Artifacts	Secondary Artifacts	Tertiary Artifacts
One dimensional	★★★	★	
One dimensional	★	★★★	★
One dimensional	★	★★	★★
One dimensional	★	★★	★★★

Table 1: Network-based detection efficacy of signature systems utilizing network behavior artifacts extracted from previously caught and classified malware.

Supervised versus unsupervised learning

The processes involved in *n*-dimensional signature development are commonly referred to as *supervised learning*, which is a category of machine learning.

Specially configured machines and environments apply numerous mathematical and statistical models to a series of labeled datasets (e.g. network traffic in the form of PCAP data from thousands of malware and benign software products).

This automatically identify which features are most pronounced between the various classes of labeled data and helps to create a signature that best aids in the differentiation of the datasets. The *n*-dimensional signature is then tested and trained against other labeled and unlabeled datasets over time to further improve the efficacy of the signature.

Another form of machine learning that is increasingly being deployed at the network-level to aid in the detection of malware threats is *unsupervised learning*.

Unsupervised learning can be best understood as a special form of anomaly detection. The machine uses a number of chosen statistical models to learn what an environment looks like and how it typically operates. This initial learning period is often referred to as *baselining*.

In the context of network-based threat detection, the machine learns which hosts and systems within the network talk to each other (e.g. how frequently they talk, what protocols they talk, and what volumes of data they exchange) and automatically identifies any notable changes in these communications.

Depending upon the sophistication of the machine and the thresholds or tolerances it is configured for, different classes of detections and alerts are possible. For example, it is relatively easy to see when a malware-infected host within the network begins to probe other hosts and begins to brute-force access credentials against another host it has not previously communicated with.

While supervised learning models are exceptional at identifying anomaly-based threats within a monitored network, they struggle significantly with labeling a threat. Without labeling a threat, it is impossible to perform any level of attribution of the threat and consequently, it is much more difficult for network administrators to remediate.

It is for this reason that supervised learning models are ideally used in conjunction with signature-based detection systems. Signature systems are used to aid the attribution labeling (and classification) after an anomaly-based threat has been detected.

Conclusion

After decades of battling the malware threat and the innovation that has been brought to bear against it, there continues to be a gap that will never be closed. Network-based detection of the threat has advanced and is the most efficient means of identifying malware-based breaches that are used to further attack the victim's network.

However, as the fidelity of network-based detection systems improved, they have evolved further away from the prospect of easily naming the type or family of threat, or from providing meaningful attribution.

While it may be a simple task to identify a paper boat floating down a river as it merges with the ocean, it is a herculean task to work backwards and (post-detection) identify the specific tributaries, streams, creeks, and springs in the mountains from which the boat was originally launched – if even possible at all. The same problem exists with network-based malware detection.

The invariances of the infection lifecycle, the malware analysis timeline, and the shifting sands of the domain name and IP timelines each combine to make correlation very difficult.

Where correlations are possible, they require huge amounts of data and a long history of observations, none of which can be reasonably harnessed at the time of the threat's detection and are circumstantial at best. As such, attribution to a specific malware or cybercriminal entity is a formidable and nearly impossible task.

The most effective methods of attribution revolve around skilled social engineering of the threat actors, and access to the logs or real-time data interception of the servers the criminals actually use to conduct their business.

Attribution is only really useful if the victim anticipates prosecuting the criminal and is prepared to spend considerable financial and reputational resources to overcome country jurisdictional issues and laws.

In rare cases where international law enforcement has been involved in building a legal case and obtaining proof of crime attributed to the malicious actor, final arrest and prosecution often takes many years – and in some cases, over a decade.

In the meantime, the malware ecosystem continues to evolve swiftly. With each new technological innovation or market opportunity, a new batch of cyber criminals look to monetize and benefit from it.

For example, the harnessing of Bitcoin payment systems as a means of monetizing new ransomware software, and the evolution of online customer services can translate the language of ransom requests and provide walk-through-guides in using Bitcoin exchanges.

The industry often goes to great lengths in labeling a new threat "advanced." It is perhaps unfortunate that most malware is already capable of automatically and remotely performing any task necessary on the victim's computer.

There is little room for more advanced features in the software components of malware. Nothing more advanced is necessary. Any advances lie in the cunningness of the attackers and the modes of monetization they will harness in the future.

Network-based threat detection focuses on the identification and classification of threats that perform network actions. The behind-the-scenes analysis of malware samples, artifact extraction, feature classification, and threat labeling are used to increase the efficacy of signatures or systems capable of performing a detection – not necessarily a tool for retroactive attribution of an in-progress attack.

It is easy for a victim to become lost in the world of attribution rather than remediation. The most efficient and precise detection methods used today lie at the network-layer, yet they make attribution a statistical curiosity, assuming there is some historical evidence for correlation.

Future advances in network-based detection systems will increase the fidelity and coverage of malware threats, but will likely move further from the possibility of specific attribution.

VECTRA® Threat Labs

Email info@vectranetworks.com **Phone** +1 408-326-2020 www.vectranetworks.com