



Suggestions for a Malcode Threat-Analysis Process Model
infectionvectors.com
Summer 2004

Introduction

The foundation of process models of any flavor is the existence of a repeatable, improvable set of procedures for accomplishing a task. Taking control of the process allows for improved efficiency and quicker troubleshooting. Within most organizations, there is a need to evaluate malicious code (malcode) on a daily basis. This evaluation must include a number of variables, including the nature of the virus or worm and the vulnerabilities of the enterprise network.

Process models that encompass the security posture for an entire organization, although exceptionally well defined in many cases (see link to OCTAVE[®] site below), cannot focus on specific threats. No matter how much detail a strategic approach has, these types of models cannot predict the types of attacks or technical vulnerabilities that may affect a network. For specific virus and exploit discoveries, it is necessary to plug a separate tactical analysis into the organization's security planning and assessment strategy. The model suggested in this paper is designed to be a guideline for anyone charged with managing security policy or incident response.

Malcode Threats

Each day, new threats are released onto the Internet. Vulnerabilities are discovered in operating systems, applications, and appliances. The ensuing reports and proof-of-concept code become part network-aware worms and root kits with increasing quickness. The ever-evolving nature of viruses makes them moving targets when it comes to threat evaluation. Many of these viruses are quite dangerous to virtually all networks and companies. However, a great deal do not present any new infection mechanisms or improvement over viruses already blocked. Each organization has its own particular vulnerabilities and strengths. As such, there should be a system in place to evaluate specific threats in the context of a particular private network.

Procedure

Receive Validated Report of New Code

These will likely come from the anti-virus vendor of choice for the network, although some groups also have in-house honeypots that will capture new threats. Possible sources of reports are major AV vendors: Symantec, Trend Micro, Computer Associates, Sophos, Panda Software, etc.). This report identifies the code type (that is, a virus, worm, macro, etc.) and provides a name for the malcode under investigation. This entry data can be

further defined by a security team to only apply to those viruses that obtain a certain level of threat, such as a “Medium” or “High” from the respective security vendor.

Identify Propagation Mechanism

Upon reception of the report, the propagation mechanism of the code will be verified. It is critical to note how the virus enters networks and computers to begin planning a mitigation strategy. Examples of propagation mechanisms are:

Email (as in the case of mass mailers)

File shares (such as worms that copy themselves to local/network directories)

Program/EXE infection (traditional virus propagation)

Buffer Overflow (those that exploit specific unchecked buffers)

File infection (such as macros)

Identify the Organization's Vulnerability to Virus' Propagation Mechanism

Based on the findings above, assign a threat level (can be in very basic terms such as “Low, Medium, & High”) to the worm based on the organization's specific vulnerabilities. This will include the patch level (in the case of buffer overruns and software bugs), the current email block list (in the case of mass mailers), and certainly the existence of any generic anti-virus signatures that will catch the code. Based on this, there may be no action required.

Identify the Impact of the Malcode

This involves assigning a value (again, this can be very simple terms) to the “worst case scenario” that the malcode produces. This is important as it will shape the amount of effort (both in technical and procedural resources) expended on the threat. This should include the costs of cleaning the virus, the value of any compromised data, the network impact, etc.

The impact of the virus can be accurately quantified only if the process of identifying and assigning values to the organization's assets has been successfully undertaken. This is an area where existing models are of great assistance.

Decide if Additional Mitigation is Necessary

Each organization will have a unique tolerance for risk and valuation of the damage caused by a virus. There is no magic number for every organization; discussions prior to an urgent threat can help define what level of risk everyone is aiming for, however. If the existing infrastructure (in terms of gateway, network, and host defenses) does not provide the level of mitigation required, it is necessary to continue with the process. In contrast, it is quite possible that the new virus or variant does not pose a unique danger to the network and is sufficiently deflected by status quo practices.

Identify Mitigation Mechanisms

Based on the propagation mechanism and consistent with the level of effort required, identify possible means of mitigating the threat. This includes email blocks, port blocks, cutting off specific commands, etc.

Order Mitigation Mechanisms

Again, the level of the threat and the possibility of infection temper these plans. The list is ordered based on a combination of how disruptive the action would be to normal traffic flow and how effective the action would be at stopping infections. This can be accomplished with a simple scale of 1-5 or descriptors such as “Very,” and “Minimal.”

Select Mitigation Appropriate to Impact

The rate of infection (propagation effectiveness) and the impact of infection will dictate the levels of disruption tolerable and the amount of effectiveness required in the mitigation efforts. Selection should be governed by

The actual implementation will follow the necessary implementation procedures (approval, testing, etc.) as time allows. Once in place, the success of the efforts should be evaluated. The success or failure of the efforts will determine whether more action is warranted.

Evaluate Mitigation

The mitigation efforts need to be judged for effectiveness. Often, just a few tests are required to ensure that a port is blocked, and extension is in the attachment drop list, etc. This may also involve using a vulnerability scanner to check that a specific patch is applied.

Finally, the once the evaluation is complete, an estimated removal timeline can be created. Often, the results of the risk alleviation will produce lasting security policies (which are evaluated in the formal change control process of the organization), other times, the mitigation efforts negatively impact the efficiency of the group and should be removed once the threat can be mitigated in another fashion (such as through client anti-virus software, once the latest signatures are available).

Exit the Process

Once the virus threat has been alleviated, this process concludes. Any outstanding efforts that are required (because of failures in the testing) will follow the same set of steps.

Implementation

This model assumes the support of multiple groups within the organization. There may already be an infrastructure for change control in place that will fit this requirement very well. The groups involved, at a minimum, are Information Assurance/Security, Network Operations (including specific areas, such as messaging, as the threat dictates), and company management.

This model may appear cumbersome when viewed against the multitude of malicious programs released each day. However, it is not meant to have a static scale, that is, many steps (especially later steps) will be eliminated for most viruses. In addition, the entry data can be more limiting (as is indicated above) to reduce the number of times the process is invoked.

Security administrators follow many of these steps informally on a daily basis. The malicious code expert on the security team may be charged with completing this process for most viruses, presenting only those with particular risk to the entire team.

Network Security Process Models

The Malcode Process Model is only one piece of a complete assessment and improvement system. There are a number of foundations for security evaluation; the most complete is a product of the SEI.

OCTAVE® <http://www.cert.org/octave/>

An extremely well-defined process for evaluating the security risks (and remediation efforts) of an organization designed by Carnegie Mellon's Software Engineering Institute (the group responsible for CMMI).

Appendix: Example Scenario

The following example outlines how the analysis and mitigation process would work given a scenario at a medium sized company. The two viruses selected were the cause of tremendous panic in early 2004:

The precursors to the notification of a new virus/worm should involve following a process that defines and values internal assets. This inventory will be used to measure the possible impact of a viral outbreak within the organization. For the purposes of this example, it is assumed that the ABC Corporation (a fictitious company of 1000 employees that provides security audits and analysis) maintains a high-visibility web server and a database that maintains customer information. Both of these information assets are valued as “mission critical,” their compromise or lack of availability would be devastating to ABC Corp. In addition, the client workstations are vital to doing business and are grouped as a collective asset. They are not critical, as they could lose a high percentage of the clients for a few days without irreparably damaging the company. In a true identification and valuation, many additional assets would likely be considered, however, for the purposes of this brief example, these are the only assets considered. Every organization has items such as the ABC web server and client database, those items that, if lost, would cripple normal operations.

The assets in question:

Item	Significance	Value	Technical
Web Server	Company Store Front	Mission Critical	Windows 2000\IIS
Database	All Customer Data\Privacy	Mission Critical	Solaris\Oracle
Clients	Productivity	Important	Windows XP

Starting with this simple foundation, the process is employed for two different threats: the MyDoom outbreak and Sasser, each considered before anti-virus signatures would be in place.

MyDoom*Report Received*

The company uses their anti-virus vendors support to receive email alerts for new viruses. In addition, the security team monitors different sites for virus reports. They get an alert for MyDoom warning that it is spreading rapidly and carries a backdoor component allowing for remote control of the compromised machine.

Infection Vectors

Quick analysis shows this is a mass-mailing worm, propagating through email attachments with an “EXE,” “PIF,” “SCR,” “CMD,” “BAT,” or “ZIP” extension. In addition, it writes to KaZaA file shares.

Exposure Level

The company has no mail client installed on the web server and prevents web surfing (and possible web-based mail clients) from either server. There is no visible file share on either device. The database server is Solaris-based.

Client workstations are behind a mail relay that strips attachments with any of extensions listed above, with the exception of .ZIP. Web-based mail account access is allowed. KaZaA is banned by policy.

The port used by MyDoom's backdoor (TCP 3127) is blocked inbound on the company's firewall.

Only the internal mail server is allowed to transmit outside of the organization (through the firewall) to TCP port 25.

Effects of Infection

The magnitude of this mass mailer is no worse than any other that ABC has evaluated. The emails the virus generates would not leave the company if a client were infected. The likelihood of infection, even without the specific signature for the client anti-virus software, is moderate (due to the use of personal web-based mail accounts). If a machine were infected, the backdoor would not be accessible to attackers.

Mitigation Required

Based on the probability of infection and the minor impact, the company has decided to send out a user alert via email, and will block .ZIP files from entering the network as email attachments until the signature for MyDoom is released.

Identify Mitigation Tools

The email gateway is the only location where attachments can be stripped on their way into and out of the network.

Order of Mitigation Efforts

See previous step, placing the extension .zip into the attachment drop list is the only action.

Action

The messaging team implements the block as per the direction of the security and management teams.

Evaluate Mitigation & Exit

A few tests from external sources prove the blocks are working. The company has set a 24-hour timeline for implementing the new virus signatures and removing the block.

Sasser

Report Received

An alert comes in for a new worm exploiting the Microsoft LSASS buffer overflow vulnerability.

Infection Vectors

The network-aware worm spreads rapidly via TCP port 445.

Exposure Level

Unless introduced intentionally, the clients are protected (TCP 445 blocked at the firewall). The Solaris database is not threatened. However, the web server running Windows 2000 could not be patched with MS04-011 due to an incompatibility between part of the patch and some proprietary software on the machine. However, TCP 445 is blocked from the outside at the company's external router. Only internally connected devices could infect the server.

Mitigation Required

The network team will verify the port 445 blocks, however, no additional steps required.