VeriSign

iDEFENSE
A VeriSign Company

IDS Evasion Techniques

David D. Rude II and Jayson Jean

drude@idefense.com

jjean@idefense.com

Where it all comes together.

Agenda

+ iDefense Overview
+ Introduction to IDS Evasion Techniques
+ Basic Evasion Techniques
+ Complex Evasion Techniques
+ Solutions
+ Conclusion
+ Q&A

About iDefense: Overview

- iDefense, a VeriSign Company, is a leader in cyber threat intelligence.
- Industry-Leading Service Offerings
  - Intelligence is all that iDefense does
- Marquee Customer and Partner Base
  - Government, financial services, retail, telecom and others
- Experienced Intelligence Teams
  - iDefense Labs
  - Vulnerability Aggregation Team (VAT)
  - Malicious Code Team (Malcode)
  - Threat Intelligence Team
  - Rapid Response Team
- In business since 1998, iDefense became a VeriSign Company in July 2005

iDefense Intelligence Services

Daily / Hourly Research Deliverables

+ Comprehensive Vulnerability Feed
  - Most comprehensive, timely, technical feed in the industry
+ iDefense Exclusive Vulnerabilities
  - More than 250 contributors around the globe
  - Released to vendor and iDefense customers only
  - More than 180 iDefense Exclusive vulnerabilities in 2005
+ Malicious Code Research and Reporting

iDefense Intelligence Services

Weekly / Semi-Monthly Research deliverables

+ Weekly Threat Report
  - Weekly compilation of worldwide threats
  - Critical Infrastructure, State of the Hack, Cyber Crime, Terrorism and Homeland Security, Global Threat
+ Bi-Weekly Malicious Code Review
  - Summary of previous two weeks malcode activity
  - In-depth analysis of specific malcode from the Malcode Lab
+ iDefense Topical Research Papers
  - Examples
    - Security of Enterprise Web-Based E-Mail Interfaces
    - Security Comparisons: Internet Explorer vs. Firefox
    - Phishing and Pharming: A Comparison
    - Mitigating the Threat from Keyloggers
+ Focused Threat Intelligence Reporting
  - Topics specific to individual customers

IDS Evasion Techniques - Introduction

- History of IDS Technology
  - IDS became popular in the mid-to-late 1990's
  - Systems were developed that monitored network traffic and compared it to known attack signatures (still commonly used technique today)
  - Snort 0.96 was released in April 2003 as one of the first open-source IDS systems
- History of IDS Evasion
  - Early evasion techniques were crude; these included generating numerous false positives and DoS attacks
  - Evasion techniques gradually became more complex and allowed little to no noise from the IDS
  - New shellcode techniques were created with the main goal being evasion of IDS
- The Race Between Developers/System Administrators and Blackhats
  - With IDS evasion techniques improving, developers and system administrators had to take precautions to prevent evasions
  - Superior stream reassembly engines were created to analyze data from the wire more efficiently
  - Normalization and Unicode processing were other early improvements
  - IDS evasion techniques have continued to improve

IDS Evasion Techniques – Basic Evasion

+ Pattern Matching Weaknesses
  - Patterns are used to match common attacks against network traffic
  - The input can change and an attack may still succeed
  - Some vulnerabilities can be triggered in different ways
  - Utilizing Unicode and URI encoding can create masked forms of web-based attacks
  - Patterns may detect some attacks but also may miss variants of the same attack

+ Unicode Evasion Techniques
  - Some IDS systems handle Unicode improperly
  - There are two different versions of Unicode (one that is outdated)
  - Unicode allows multiple representations of the same characters (old standard)

IDS Evasion Techniques – Basic Evasion

- Denial of Service (DoS) Attacks
  - Many IDS systems log alerts to a central location
  - Central log servers may be a target for DoS attacks
  - Filling disk space with false positives (time consuming)
  - Slow down IDS processing time
- False Positive Generation
  - Generating a large number of false positives may help mask the real attack
  - An attacker needs to have a general idea of rules used on the IDS
  - The more rules that are known, the better the chances of masking the real attack
  - Many IDS systems come with default rules that are similar from vendor to vendor

IDS Evasion Techniques – Basic Evasion

- Session Splicing
  - IDS Systems did not always reconstruct sessions before doing pattern matching on the traffic
  - Splitting data among several packets to make sure each packet does not match any patterns will bypass detection
  - Reassembly timeouts may also come into play
  - If an application will wait for input longer than the IDS is willing to reassemble, detection may be avoided

IDS Evasion Techniques – Complex Evasion

- Fragmentation
  - Two methods are commonly used to evade IDS:
    - Method One: overwrite a section of a previous fragment.
    - Method Two: overwrite the complete fragment previously transmitted.
  - Methods that are used can be intermixed
  - Modern IDS are able to handle this type of attack
- Time to Live
  - Relies upon knowing distances between the end host and the network IDS
  - Injects packets into the IDS stream without letting these packets reach the end host
  - The IDS may not realize that this particular packet did not reach the end host
- Invalid RST Packets
  - By sending an invalid RST packet, the IDS may stop processing the stream and the end host will ignore it
  - Uses invalid checksums (there may be other ways)

IDS Evasion Techniques – Complex Evasion

- Urgency Flag
  - RFC 1122 “1 Byte data, next to Urgent data, will be lost, when Urgent data and normal data are combined.”
  - Similar to Fragmentation Method One (Overlap)
  - Insertion of one byte within a pattern may bypass IDS detection
  - IDS systems may improperly handle the urgency flag
- Polymorphic Shellcode
  - Is dynamic, as it uses a different encryption key each time this shellcode is used
  - Hides the content of shellcode
  - Many IDS systems have rules that look for common strings within shellcode
- ASCII Shellcode
  - Similar to polymorphic shellcode
  - Does not change dynamically
  - Hides shellcode content

IDS Evasion Techniques – Complex Evasion

+ Application-Layer Attacks
  - Application-Layer data is often compressed (zipped)
  - Attacks can be sent within compressed data
  - Sometimes there are multiple triggers for a vulnerability
  - Integer overflows can use different values to reach the same goal
  - Shellcode can be hidden in different ways

IDS Evasion Techniques – Solutions

- Solutions
  - There is very little that users can do to limit evasion
  - The best approach is to maintain security awareness and patch regularly
  - Choose an IDS system wisely based on network topology and network traffic
  - Look for features within an IDS that will help prevent as many evasion techniques as possible
- Normalization
  - Translates obfuscated input into what the end host will eventually see
  - Usually applied to Unicode, UTF8, URI encodings
  - Attempts being made to normalize polymorphic shellcode
  - Normalizing polymorphic shellcode decreases performance
  - Normalization of fragmented packets
  - Time-To-Live field ensures end host delivery

IDS Evasion Techniques – Solutions

- Packet Interpretation Based on Target Host
  - IDS tries to recreate what the end host will see
  - Many different ways that operating systems handle standards
  - Target Host TCP/IP stacks would be more accurate in detection of attacks
  - Modular TCP/IP stacks could be useful to separate end host from IDS
    - Research into this field is required to accurately write the modular TCP/IP stacks
  - May prove to be very effective in mitigating network-level evasion techniques
- Time-To-Live Problem
  - Two potential methods of dealing with this issue:
    - Method One: Increment TTL Field to a large value for every packet
    - Method Two: Actually trace the network topology by mapping MAC addresses to distance and using this map to detect packets that will not reach the end host

IDS Evasion Techniques – Solutions

+ Dealing With The Shellcode Problem
  - Polymorphic shellcode detection based on a nop opcode count threshold
  - Numerous nop codes are used within polymorphic shellcode to mask it from the IDS
  - Known to create false positives

IDS Evasion Techniques – Conclusion

- Conclusion
  - IDS technology needs to detect all attacks and mitigate evasion techniques in order to be truly effective
  - IDS has it's limitations, but can be a powerful tool
  - Reinforce IDS with other technologies that compliment it
  - Limit the avenues of attack with sound security practices
  - When deploying IDS, consider things like network topology, network traffic, number of hosts on the network and points on the network that will require monitoring
  - Remember to choose an IDS system wisely based on its features and abilities to mitigate evasion techniques

Q/A

David D. Rude II and Jayson Jean

iDefense Intelligence Operations

iDefense, a VeriSign Company

Where it all comes together.