WG: Requirements for Network Monitoring from an IDS Perspective

Lothar Braun, Falko Dressler (chair), Thorsten Holz, Engin Kirda, Jan Kohlrausch, Christopher Kruegel, Tobias Limmer (note taker), Konrad Rieck, James Sterbenz

Challenges

Detection of malicious traffic is based on its input data, the information that is coming from network-based monitoring systems. Best detection rates would only be possible by monitoring all data transferred over all network lines in a distributed network. Monitoring and reporting this amount of data are feasible in neither today's, nor will be in future's systems. Later analysis like stateful inspection of the traffic imposes even more processing costs. But only at this level of monitoring and analysis there may be a chance to capture all attacks inside a system. So there needs to be a trade-off between detection success and the processing costs.

Malicious traffic is mostly generated by compromised systems. Catching attackers during the process of taking over a vulnerable host is complicated, as such attacks only use very low traffic volumes. Operating system security improved in the last years, as because of stack protection and firewalls security holes are more difficult to exploit. To counter this problem, more attacks will base on social engineering techniques like phishing or spam mails. Methods need to be developed to detect those kinds of attacks.

For higher monitoring data rates, the effects of attacks may be easier monitored, like scans for vulnerabilities, large file transfers or mass spam distribution. Most of those attacks are initiated by script kiddies using downloaded tools, maybe slightly modified. So far, these automated attacks focus on the mass market and do not implement any sophisticated anti-IDS techniques. This fact improves the chance of detecting these attempts dramatically.

The location of monitoring systems also poses several unsolved challenges: monitoring and analysis can be run in a combined way on end systems. This way, the attack detection is heavily distributed and would enable full payload inspection, although in this solution correlation of the analysis results will be difficult. The more conventional way of placing monitoring systems on the network backbone implies that high data rates only allow a coarse analysis of the data. Detailed analysis would only be feasible for a portion of the traffic. Adaptive monitoring would allow attack detection algorithms to select suspicious data for more detailed inspection in this scenario. It will be easier for attackers to avoid detection by coarse anomaly-based algorithms than detailed inspection, but evasion will always be possible unless detection methods analyze full packet payload data and detect anomalies in the semantic content of the application layer in the exchanged protocol data. The tradeoff becomes visible in IP and TCP fragmentation issues: connection reassembly does not offer the

Dagstuhl Seminar Proceedings 08102
Perspectives Workshop: Network Attack Detection and Defense
http://drops.dagstuhl.de/opus/volltexte/2008/1497

speeds required for the data rates that occur in backbone networks. Essentially, methods are needed that reduce the traffic obtained in network monitoring tools. One idea is the use of black-/whitelists which define what data is benign and does not need to be monitored and classified. Metadata could solve this problem, like the case of trusted systems that supply information about transferred traffic to the monitoring system and tells what parts of the traffic is benign.

So far we only considered plain, unencrypted traffic whose payload may be analyzed directly. But current trends in networking show that the amount of encrypted traffic, tunnels and, in general, overlays is increasing. Monitoring this data introduces more problems: Content is obscured and only statistical features may be used for detection of malicious data. Usage of other networks like 3G networks for mobile devices also increases and those offer entire new types of attacks, as new papers about e.g. power depletion attacks show. The structure of these networks is often fundamentally different from the internet: the operator has complete control over the network. As only little information is available about those networks, we did not include them in our considerations.

Assessment of state-of-the-art

Attacks

The detection efficiency of a network based intrusion detection system heavily depends on the type of attacks and the intention of the attacker. The statistics of computer security incident response teams (CSIRT) show that non-targeted attacks exploiting either well-known vulnerabilities (e.g. the statistics of CERT Polska [1]) or applying social engineering techniques are by far the most common attacks seen in the Internet. The compromised machines are typically integrated into a botnet ([2]) and abused for phishing fraud, attacks on other machines or sending spam emails. For example, vulnerabilities on out-dated versions of the Microsoft Windows operating system are still exploited to compromise vulnerable systems (the survival time is predicted in [5]). An example for social engineering attacks is the "storm worm" ([3]) which spreads through email attachments. The content of the emails are adapted to topics of actual public interest (originally, the topic of the mails was related to the storm "Kyrill"). Another important class of attacks tries to exploit vulnerabilities in web server on the application layer like PHP code injection or SQL injection vulnerabilities. Common to these vulnerabilities is the trivial way in which both can be leveraged to take control over the machine. For example, the Joomla/Mambo CMS and a very large number of other PHP application suffered in the past from these vulnerabilities and it can be expected that a multitude of other similar vulnerabilities will be found (e.g. see [4] for public exploits on Joomla/Mambo).

Netflow monitoring

Netflow monitoring has become widely accepted as standard to create statistics about network traffic transferred by routers. The IETF has defined a protocol to carry flow information over the network called IP flow information export (IPFIX). Included is a standard which allows the transfer of per-packet information like payload data. Standard 5-tuple flow aggregation produces around 8000 flows/s for a 1 GBit/s link. Based on a hash table, only few methods are available for mitigating DoS attacks on monitors, one of them can be found in [Hu06]. To enable processing of high data rates, dynamic reconfiguration of monitors become an issue for attack detection. At the moment, only few monitoring systems support seamless reconfiguration without packet losses. To solve the problem in environments with even higher speeds, several methods of packet sampling and filtering are employed [estan2004].

Recommendations/conclusions

How to get started

Pure monitoring and analysis of flow data is not sufficient for currently available detection algorithms. A trade-off would be the use of flow data labeled with payload information, like the first N byte of a stream. This solution would be challenging, but feasible. Distributed Denial-of-Service attacks can be countered with intelligent hashing and intelligent sampling inside the network monitors to avoid complete failure of the systems.

Future internet architectures

In today's internet protocol architecture the application layer protocol number (i.e. destination port) is meaningless. Furthermore, the specification of the lower protocol layers allow "interesting" protocol use like using only one byte of payload per packet.

These limitations can be countered in the future: The introduction of a separate control plane which allows application layer protocol checks by lower layer processing entities would lessen this problem. IANA could assign IDs for protocols that have a defined specification also involving lower layer packet structures. An example of this idea could be that a specific data request (e.g. a HTTP request) must lie inside the first one or two packets of the corresponding connection. Connection reassembly would not be needed any more. The challenge for this problem is it to be designed properly without limiting the flexibility of having stacked protocol layers. The first step of starting this development could be the implementation of application layer checks and lower level protocol restrictions in currently used protocols, so that both approaches remain compatible to each other for easy migration. The next step would be the definition of a new clean-slate approach for future protocols.

From a networking perspective end-to-end communication in current networks is state-of-the-art, although developments in the internet of the last years show a trend to NATed networks. From the security perspective, the introduction of NAT routers is a good idea, as hosts inside these networks have a better protection. This protection could be offered by ISPs in the future, but it would require high costs for managing the infrastructure. Furthermore, some networking methods like peer-to-peer protocols are blocked out by this approach. Full security can almost never be offered, as can be seen in new attacks focusing on compromising NAT routers or client hosts inside the LAN.

As a reaction to the detection of compromised hosts, those systems may be placed within a safe, controlled domain by the provider which confines further spread of the infection, notifies the user of the compromise and offers remedies. It is not clear, if this approach is feasible outside a well-controlled network such as a university network. Investigations into user-friendly solutions to this problem are definitely required.

References

[1] CERT Polska / Arakis incident statistics / URL: http://www.arakis.pl/en/tbs.html

[2] "Know your Enemy: Tracking Botnets" / Paul Bächer, Thorsten Holz, Markus Kötter, Georg Wicherski / URL: http://www.honeynet.org/papers/bots/

[3] Symantec.com / W32.Storm.Worm / URL: http://www.symantec.com/security_response/writeup.jsp?docid=2001-060615-1534-99

[4] Milworm exploit archive / http://www.milw0rm.com

[5] Internet Storm Center/ http://www.isc.sans.org/survivaltime.html

[6] Hu, Y.; Chiu, D. & Lui, J. C. (2006), Adaptive Flow Aggregation - A New Solution for Robust Flow Monitoring under Security Attacks. In: IEEE/IFIP Network Operations and Management Symposium (IEEE/IFIP NOMS 2006), pp. 424-435.

[7] Estan, C.; Keys, K.; Moore, D. & Varghese, G. (2004), Building a better NetFlow. In: 'ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '04), pp. 245-256.