RAID 2006 - Recent Advances in Intrusion Detection
Summary and Comments

Matthias Vallentin
Research Unit VIII: Network Architectures
Computer Science Department
Technical University Munich

Hamburg, Germany, September 20 – 22, 2006
Conference Overview

Invited Talk

Sessions
- Anomaly Detection
- Attacks
- System Evaluation and Threat Assessment
- Malware Collection and Analysis
- Anomaly- and Specification-Based Detection
- Network Intrusion Detection

Summary
RAID - Recent Advances in Intrusion Detection

**Topics**: intrusion detection and defence

**Annual series**: 9th

**Location**: Hamburg, Germany

**Targeted audience**: leading researchers, academia, government, and industry

**Intention**: promote and further advances in intrusion detection.

**Submissions**:
- Full Papers
- “Big Challenge” and “Big Idea” – fundamental problems, not yet tackled
- Panel Proposals – discussing hot topics
- Poster sessions – work in progress
Selected List of Topics

- Risk assessment and risk management
- Intrusion tolerance
- Deception systems and honeypots
- Vulnerability analysis and management
- IDS assessment, survivability, and integration
- IDSs in high-performance and real-time environments
- Privacy aspects
- Vulnerabilities and attacks
- Data mining techniques
- Case studies
- Legal and social issues
Invited Talk

- **Author:** Joseph Springsteen
  Computer Crime and Intellectual Property Section
  U.S. Department of Justice
- **Field:** Intersection of Computer Security and Law Prosecution
- **Title:** *The New World of Networked Cybercriminal*
- **Roadmap:**
  - Trends in computer network crime
  - Anatomy of a computer crime: carding
  - Lessons learned about the new networked cybercriminal
More than 10,992 new Windows viruses and worms in the last half year of 2005.

More than 31% of IP source addresses linked to attacks are from the US, followed by China (7%), UK (6%), and Germany (5%).

US has most bot-infested computers (26%), followed by the UK (22%), China (9%), and France (4%).

Definition

A **botnet** is network comprised of infected machines (**zombies**, **drones**, or **(ro)bots**) that can be remotely controlled by an attacker.
Increasing attacks against web-based applications.

- Message forums
- E-commerce (e.g. shopping carts)
- Blogs
- Webmail

Browser vulnerabilities constantly grow.
Hacker Motivations

- **First generation**: scientific or intellectual curiosity
  - *Morris* Worm
- **Second generation**: hacker “cool” or *DefCon* wannabees
  - Web page defacements.
- **Third generation**: Hacking for profit
  - Botnets
  - Carding
  - Phishing
  - Hackers for hire
Hacker Targets

- Financial institutions
- Education
- Small business
- E-commerce sites
- Government / Military
- Home Computers (for the bot army)
Code is “Root” of the Problem

Code characteristics leveraged by malware writers:

- **Complexity**
  - High # of lines of code (LOC)

- **Extensibility**
  - Updates
  - Extensions
  - Modularity

- **Connectivity**
  - Ubiquity of the Internet
  - Multiple attack vectors on the clients (mail clients, browsers, etc.)
853 “high severity” vulnerabilities disclosed in last half of 2005.

Exploit code developed and published an average of 6.8 days after the announcement of a vulnerability.

49 days to issue a patch (down from 64).
Operation Firewall is a project from the U.S. Department of Justice to counter:

- Hacking / network intrusion
- Spam, phishing, trojans
- Identify theft
- Credit card fraud
- Financial institution fraud

Focus lies on organized criminal groups.
Centers of Online Crime

Two facets of online crime:

- **Financial crime**
  - Credit Card (CC) fraud
  - Bank fraud and money laundering
  - Identity theft

- **Computer crime**
  - Hacking services
  - Custom malware
A Primer on Carding

Some vocabulary:

- **Carder**: criminal specialized in carding.
- **Dump**: information encoded on the back of a credit card.
- **Cashing out PINs**: illegally obtaining funds from an ATM through the unauthorized use of bank account information encoded onto counterfeit plastic cards.
- **Carding online**: using stolen account numbers to make Internet purchases.

What can you purchase?

- Financial account information, CCs, passports, driver’s licences, birth certificates, social security cards, credit reports, insurance cards, diplomas, . . .
Lessons Learned: The new Networked Cybercriminal

- Change in hacking culture
  - Emergence of the profit-seeking, sophisticated online fraudster
  - More intellectual and skilled hackers
- Development of new type of organized criminal activity
  - Highly organized and structured, but loosely affiliated membership
  - Increasing number of organized criminal groups
  - Not yet “organized crime”
- Prototype: young, early 20s, male, loner
The Future: Law Enforcement Needs

- Improvement in incident reporting
- Better documentation of damage and loss associated with intrusions
- The value of live incident response
- Continued technological advancements: hackers lead the arms raise
Outline

1. Conference Overview
2. Invited Talk
3. Sessions
   - Anomaly Detection
   - Attacks
   - System Evaluation and Threat Assessment
   - Malware Collection and Analysis
   - Anomaly- and Specification-Based Detection
   - Network Intrusion Detection
4. Summary
Session Talk 1 of 2

Paper

James J. Treinen and Ramakrishna Thurimella. *A Framework for the Application of Association Rule Mining in Large Intrusion Detection Infrastructures.*
Motivation

The problem: large intrusion detection infrastructures

- > 1,000 sensors
- 1,000,000 – 10,000,000 alarms/day
- 99% false positives

*Meta signatures or rules* identify known attack patterns, **but**

- new rules appear almost daily
- maintaining the rule base manually is time consuming and error prone
Contributions

- Novel approach using *association rule mining* to shorten the time to incorporate new attack profiles into the production infrastructure.

- Ability to flag suspicious network activity for in-depth analysis to detect attacks which may else have been lost in the large data volume.

Conclusion

- The Framework reduces the amount of manual inspection required to generate new rules.

- decreases the time of the process from discovery to implementation of new rules.

⇒ reduces the overall cost of maintenance.
Debin Gao, Michael K. Reiter, and Dawn Song. *Behavioral Distance Measurement Using Hidden Markov Models.*
Definition

The *behavioral distance* between two processes is a measure of the deviation of their behaviors.

Motivation

- An increase in behavioral distance might indicate the compromise of a process.
- Central research challenge: constructing a robust model with respect to *mimicry attacks*. 
Example

1. Mimicry attack modeled as a sequence of system calls.
2. Measuring behavioral distance on a replicated service by inserting bogus system calls to undermine the model.
3. Compute behavioral distance on a
   - different code base, replicated service (Apache ⇔ Abyss).
   - different platform (Windows ⇔ Linux).
4. Deviations in the behavior of processes on the same input are indications of intrusions.
Contributions

- New approach to behavioral distance calculation using a new type of Hidden Markov Model.
- Evaluation the capability of the proposal when used to measure the distance between the system-call behaviors of diverse web-servers.

Conclusion

- A system-call-based anomaly detector for host-based intrusion detection.
- New algorithm for computing behavioral distance between processes.
Outline

1. Conference Overview
2. Invited Talk
3. Sessions
   - Anomaly Detection
   - Attacks
   - System Evaluation and Threat Assessment
   - Malware Collection and Analysis
   - Anomaly- and Specification-Based Detection
   - Network Intrusion Detection
4. Summary
Jonathon T. Giffin, Somesh Jha, and Barton P. Miller. *Automated Discovery of Mimicry Attacks*. 

Session Talk 1 of 3
Motivation

- Model-based anomaly detection systems restrict program execution by a predefined model of allowed system call sequences.
- Previous research employed **manually** constructed mimicry and evasion attacks that avoided detection by hiding malicious sequences of system calls within a valid sequence.

⇒ **Automate** the discovery of such attacks.
Contributions

- Using model checking to find sequences of system calls accepted as valid by a program but inducing malicious effects on the operating system.
  - Revealing undetected attacks by finding counter-examples that cause the proof to fail.
  - E.g. system call sequences that are accepted as valid execution but induce malicious attack effect.
- A system design where attack sequences and obfuscations need not be known.
Conclusion

- Approach to connect attack discovery with attack construction for complete and automatic evasion of detectors.
- Leveraging automation to find undetected attacks against program models that previously were found only with manual inspection.
Session Talk 2 of 3

Paper

Simon P. Chung and Aloysius K. Mok. Allergy Attack Against Automatic Signature Generation.
Motivation

- Automatic worm containment is increasingly looked upon as a solution to filter out *0-day* worm instances.

- *Automatic signature generation (ASG)* are worm containment systems which aim to defend against worms, especially those spreading too fast for human intervention to be useful.

- "*Allergy Attack*": manipulating the system and turn it into an active agent for DoS attack against the protected system, that is, inducing ASGs into generating signatures that match normal traffic.

⇒ Normal traffic will be blocked when signatures are applied to the perimeter defense, resulting in the desired DoS.
Contributions

- Demonstrating the attack against a typical ASG system (autograph). The authors hope to draw attention to the threat posed by this type of attack.

- Presenting the insight on what caused the attack shall help the design of future ASG systems to be resilient against allergy attacks.
Conclusion

- The allergy attack is a low-cost DoS attack avieved through the manipulation of ASGs protecting the target network.
- While *polymorphism* only renders ASGs useless, allergy attacks turn them into real harm to the protected network.
- 8 of 9 examined ASGs are vulnerable to this attack.
Session Talk 3 of 3

Paper

James Newsome, Brad Karp, and Dawn Song. *Paragraph: Thwarting Signature Learning by Training Maliciously.*
Motivation

Defending against worms and defending against spam bear certain similarities. In both cases

- a stream of samples arrive,
- a classifier must determine whether each sample falls into a malicious target class,
- a learner generates a classifier automatically by analyzing two labeled training pools: innocuous and malicious samples.

A delusive adversary can manipulate the training data to prevent a learner from generating an accurate classifier, even if the training data is correctly labeled.

⇒ Severe consequence to training on data provided by a malicious teacher.
Contributions

- Definition of the classifier generation problem as a learning problem in an adversarial environment.
- Description of attacks on learning classifier generators that involve careful placement of features in the target-class training data.
- Analysis and simulation of these attacks to demonstrate the efficacy in the polymorphic worm signature generation context.
Conclusion

- Learning an accurate classifier from data largely controlled by an adversary is a difficult task.
- The problem of a delusive adversary must be taken into account in the design of malicious classifier generation systems.
- Promising approaches include
  - designing learning algorithms that are robust to maliciously generated training data,
  - training using malicious data samples *not* generated by a malicious source,
  - performing deeper analysis of the malicious training data to determine the semantic significance of features before including them in a classifier.
Outline

1. Conference Overview
2. Invited Talk
3. Sessions
   - Anomaly Detection
   - Attacks
   - System Evaluation and Threat Assessment
     - Malware Collection and Analysis
     - Anomaly- and Specification-Based Detection
     - Network Intrusion Detection
4. Summary
Session Talk 1 of 3

Paper

Motivation

Reflecting intrusion detection: is *anomaly detection* improving in efficacy and reliability?

Current evaluation strategies suffer from problems:
- they produce results only valid within the evaluation data set,
- they provide very little by way of diagnostic information to tune detector performance.

⇒ Focus on the problem of acquiring reliable performance results for an anomaly detector.
Contributions

- Identifying the detector’s blind spots and sensitivities to various forms of anomalies.
- Providing diagnostic information to explain why the detector performed well or poorly (these results extend to arbitrary data sets).
Conclusion

- Evaluation approach for an anomaly-based detector, using a parameterized family of synthetic data sets.

- Comparing the observed performance characteristics on the synthetic data with real-world data yield similar performance curves.

- Challenges remain to be faced in the area of intrusion-detector performance modeling. (e.g. multiple local optima make it difficult to find a satisfactory tuning.)
Session Talk 2 of 3

Paper

Definition

An **attack graph** is a general formalism used to model security vulnerabilities of a system and all possible sequences of exploits which an intruder can use to achieve a specific goal.

Motivation

- The size and complexity for real systems greatly exceeds human ability to visualize, understand and analyze.  
  \Rightarrow \text{Identify only relevant portions of an attack graph.}

- Introducing a ranking scheme:
  - the rank of a state shows its importance.
  - e.g. the probability of an intruder reaching that state.

- Given a ranked attack graph, the security analyst can concentrate on relevant subgraphs to deploy security measures.
Contributions

- The authors propose a ranking scheme for the states of an attack graph: Given a security property, an attack graph can be automatically generated using model checking.

- Presentation of two ranking algorithms to rank states based on the probability of an intruder reaching those states:
  1. algorithm similar to the PageRank algorithm used by Google.
  2. algorithm that ranks states based on the reachability probability of an intruder in a random simulation.
Conclusion

- Ranking of attack graphs helps in overcoming the visual complexity.
- Even if the exact probabilities are not available, modeling the attacks randomly is expected to perform as good as PageRank performs on the World Wide Web graph.
Paper

Motivation

- Security-oriented risk assessment tools are used to determine the impact of certain events on the security status of a network.

- Existing approaches are limited to *manual* and *time-consuming* risk evaluations.

- Introducing *Real-time* risk assessment based on initial risk assessment and IDS sensors:
  - alerts are prioritized according to the risk of involved hosts
  - responses are initiated at the hosts based on the assessed risks
Contributions

- Novel approach to network risk assessment: considers the risk level of a network as the composition of the risks of individual hosts.
- Using *Hidden Markov models (HMMs)* to represent the likelihood of transitions between security states.
- Real-time risk assessment tool integrated with the STAT framework.
- E.g., if a host on the inside of a network is compromised, the risk level of other hosts within the network increases as well.
Conclusion

- Attempt to quantify the risk activity of networks and individual hosts in real-time with simulated an real-world data.

- Important limitation: need for model parameter estimation. (here: manually estimated)

- Not yet tested on-line with live-traffic, only off-line examination.
Outline

1. Conference Overview
2. Invited Talk
3. Sessions
   - Anomaly Detection
   - Attacks
   - System Evaluation and Threat Assessment
   - Malware Collection and Analysis
   - Anomaly- and Specification-Based Detection
   - Network Intrusion Detection
4. Summary
Session Talk 1 of 3

Paper

Definition

A **honeypot** act as “a resource whose value is being in attacked or compromised”. It is expected to get probed, attacked, and potentially exploited.

- **Low-interaction honeypot**: only light-weight application responders (e.g. honeyd).
- **High-interaction honeypot**: emulating a complete operating system.
- **Medium-interaction honeypot**: inbetween low- and high-interaction honeypots.
Motivation

- Software with malicious attempt (malware) poses a constant threat to hosts connected to the Internet.

- A thorough understanding of the malware’s internals is essential to understand its impact and to construct adequate defense mechanisms.
  ⇒ Collecting malware in order to
    - understand and dissect malware,
    - anticipate attack trends and attack rates.

- However, hundreds of new malware species emerge every month rendering manual dissection impossible.
  ⇒ A high degree of automation is required to handle the vast amount of malware samples.
Contributions

- Introducing *nepenthes*, a honeypot platform to deploy honeypot modules based on a *low-interaction* honeypot.
- By only emulating the vulnerable part of a service, a high degree of scalability is achieved.
The *nepenthes* platform is an approach focused on large-scale malware collection.

With one physical machine, the system was able to listen to more than 16,000 IP addresses in parallel.

The platform collected 14,000 unique malware binaries thus far.
Corrado Leita, Marc Dacier, and Frederic Massicotte. *Automatic Handling of Protocol Dependencies and Reaction to 0-Day Attacks with ScriptGen Based Honeypots.*
Motivation

- High-interaction honeypots involve high maintenance costs and a substantial amount of resources.
- Low-interaction honeypots lack the flexibility to carry on the conversation long enough to distinguish between different attacks.

⇒ Enhance the expressiveness of a low-interaction honeypot by automatically generating responders.
Contributions

- An algorithm to infer dependencies in the content of protocol messages without requiring the knowledge of protocol semantics.
- An algorithm to generate relations in the interaction of multiple TCP sessions.
- A proxying algorithm that allows a honeypot to automatically build a training set to refine its knowledge of protocol reacting.

Conclusion

Protocol-independent approach to build scripts to emulate servers under attack.
Session Talk 3 of 3

Paper

Moheeb Abu Rajab, Fabian Monrose, and Andreas Terzis. *Fast and Evasive Attacks: Highlighting the Challenges Ahead.*
Motivation

- *Network telescopes*: traffic monitors located in routable but unused IP space.
  - Detect and characterize *backscatter*.
  - Detect malware.

- As the use of such monitors becomes commonplace, malware will evolve to actively detect and evade them.

- Recent malware (e.g., Agobot) already avoids well-known monitors and prefixes of certain agencies.

⇒ Highlight the threat of simple and effective attacks that undermine the usefulness of passive monitors.
Contributions

- Introducing a new lightweight sampling technique to detect dark space, i.e. unused IP space and passive network monitors.
- Demonstrate the virulence of evasive malware spreading.

Conclusion

- With a reasonably small number of probes, it is possible to
  - accurately detect the locations of passive network monitors
  - identify live address clusters containing the majority of the vulnerable population.
- The proposed design of evasive malware is capable of evading extensive collections of network monitors, while saturating the vulnerable population in seconds.
Outline

1. Conference Overview
2. Invited Talk
3. Sessions
   - Anomaly Detection
   - Attacks
   - System Evaluation and Threat Assessment
   - Malware Collection and Analysis
   - Anomaly- and Specification-Based Detection
   - Network Intrusion Detection
4. Summary
Motivation

- Current generation of *Network Intrusion Detection Systems (NIDSs)* is not able to detect
  - stealthy worms,
  - 0-day exploits,
  - polymorphic attacks employing code obfuscation.

⇒ Detect these attacks via a content/payload anomaly detector.
Contributions

- **Anagram**: A statistical content-based anomaly detector based on semi-supervised training of higher-order n-gram analysis.
- Robustness against future mimicry attacks by the use of a randomized testing strategy.
- A new defensive strategy that combines host-based and content-based sensors to improve accuracy of modeling a site’s content flow.
Conclusions

- Anagram: higher-order n-gram modeling
- Compactness and performance by using *bloom filters*.
- Randomized modeling/testing can help thwart mimicry attacks.
- Evaluation suggests that Anagram has less than a 0.1% false positive rate with 100% detection rate in network traces.
Session Talk 2 of 2

Paper

Definition

A Mobile Ad Hoc Network (MANET) is a distributed cooperative communication platform for mobile wireless nodes without a pre-established infrastructure.

- Mobile nodes act as routers to forward packets.
- “trust-all peer” design.

Motivation

- No centralized monitoring point in MANETs, usually a distributed architecture with sensors.
- Malicious nodes can easily corrupt the routing ability of the network.
- Yet no IDS has a practical message exchange to supply an IDS with data.
Contributions

- Practical IDS framework for MANET: *Distributed Evidence-driven Message Exchanging intrusion detection Model (DEMEM)*:
  - detectors intercept and validate routing messages,
  - low message overhead (→ *scalable*).
- DEMEM implementation in *Optimal Link State Routing (OLSR)*.
Conclusion

- **DEMEM:**
  - first generic IDS architecture for MANET.
  - resolve unique requirements of IDS in MANET.

- **DEMEM in OLSR:**
  - detects OLSR attacks by constraints.
  - Newly proposed ID messages supply sufficient evidence in real-time.

- **Performance:**
  - Low performance overhead,
  - very low false positives, no false negatives,
  - low detection delay.
Outline

1. Conference Overview
2. Invited Talk
3. Sessions
   - Anomaly Detection
   - Attacks
   - System Evaluation and Threat Assessment
   - Malware Collection and Analysis
   - Anomaly- and Specification-Based Detection
   - Network Intrusion Detection
4. Summary
Session Talk 1 of 3

Paper

Jose M. Gonzalez and Vern Paxson. *Enhancing Networking Intrusion Detection with Integrated Sampling and Filtering.*
Motivation

“Main Path”: most NIDS analysis entails looking at large volumes of traffic in a stateful fashion, (e.g. filter packets, reassemble packet streams into ADUs).

However, other forms of analysis does not require such deep-and-stateful processing.

⇒ Augment a NIDS’s analysis with an additional, separately filtered stream of packets (a lightweight “Secondary Path”) complementing the main analysis.
Contributions

Two enhancements to the popular Berkeley Packet Filter (BPF) that allow analyzers to “cherry-pick” packets:

- Randomness as a first-class object in BPF: in-kernel random sampling of packets, connections, hosts, or such.
- Richer in-kernel filter control mechanisms, including a lightweight form of persistent state (i.e. a mechanism for storing/recovering information across packets).
Conclusion

- Additional power through Secondary Path, e.g.
  - disambiguating the size of large TCP connections,
  - finding dominant traffic elements (“heavy hitters”),
  - enhancing backdoor detection.
Session Talk 2 of 3

Paper

Motivation

- Signature-based IDS are deployed in a large number of organizations.
- Scalability is a growing concern:
  - network bandwidth is increasing,
  - signatures are increasing in number and complexity.
- Significant effort in improving methods for efficient deep inspection.

⇒ For higher performance, IDS should adapt according to the workload.
Contributions

- A method for improving the performance of IDS system by adapting to the input rules and the observed network traffic.
- A workload-aware Snort prototype that consists of two components:
  1. **profiling** component that profiles both the input rules and the observed network traffic to produce an evaluation strategy.
  2. **pre-processing** component that filters the rules according to the evaluation strategy (and then matches incoming packets).
Conclusion

- A method for improving performance of IDS by adapting to the rule-set, network traffic.

- The results show that work-load aware intrusion detection outperforms Snort by up to 1.6 times (web-based rules: 2.7 times) and consumes 10-20% less memory.
Session Talk 3 of 3

Paper

Motivation

- The authors claim: current NIDS have a narrow scope.
  - NIDS tackle flow aggregates, TCP reassembly, individual packets or application-level data fields,
  - but no solution is capable of handling all of the above.

- Dilemma:
  - Systems handling evasion and polymorphism are too slow for in-line deployment.
  - Fast in-line solutions are not able to detect and stop sophisticated attacks.

⇒ Building a network card that can be deployed in-line at gigabit rates.
Contributions

- A practical system, *SafeCard*, capable of reconstructing and scanning TCP streams at gigabyte rates while preventing polymorphic attacks, using up to layer-7 checks.
- A high-performance pattern matching language, *Ruler*, that offers functionality similar to Snort but amenable to implementation on low-level hardware.
Conclusion

- **SafeCard** is an in-line NIDS featuring detection techniques at all levels of abstraction.
- Capable of handling close to a gigabit per second of TCP traffic.
- Implemented as a pipeline on a single Intel IXP2400 network processor.
Active research in various fields.

Promising venues for future work, but real-world applicability still has to be improved.

A global threat forces the security community to increase international collaboration.

As the sophistication of adversarial attacks increases, protection systems have to adapt in turn. The arms race continues...
Thank you for your attention and patience.