Go ahead of malwares infections and controls: Towards new techniques for proactive cyber defense

Guofei Gu
TEXAS ENGINEERING EXPERIMENT STATION COLLEGE STATION
1470 WILLIAM D FITCH PKY
COLLEGE STATION, TX 77843-3577

12/08/2016
Final Report

DISTRIBUTION A: Distribution approved for public release.
**14. ABSTRACT**

We are currently witnessing a sea change in cyber crimes carried out by malware, i.e., from fun/fame driven to profit driven. With strong economic incentives, cyber criminals are writing much better malware. Even worse, the attack-defense arms race actually favors attackers, because they are more proactive in designing/testing new attacks. To address the challenge, this research aims at designing novel proactive techniques to defend against current and next generation malware/attacks. We focus on three areas. First, we developed several new advanced malware analysis and detection techniques for both PC and emerging Android smartphone platforms. Second, we developed two novel proactive techniques for active, robust, fast and scalable malware server infrastructure detection at the Internet scale. Finally, for proactive defense, it is very important to look forward to the security issues in the next generation networking technology, which is widely believed to be Software-Defined Networking (SDN). We have discovered several new vulnerabilities in SDN and developed new defense for SDN, as well as new security capabilities using SDN.

**15. SUBJECT TERMS**

Malware, security, botnet, SDN
INSTRUCTIONS FOR COMPLETING SF 298

1. REPORT DATE. Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g. 30-06-1998; xx-06-1998; xx-xx-1998.

2. REPORT TYPE. State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

3. DATES COVERED. Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

4. TITLE. Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

5a. CONTRACT NUMBER. Enter all contract numbers as they appear in the report, e.g. F33615-86-C-5169.

5b. GRANT NUMBER. Enter all grant numbers as they appear in the report, e.g. AFOSR-82-1234.

5c. PROGRAM ELEMENT NUMBER. Enter all program element numbers as they appear in the report, e.g. 61101A.

5d. PROJECT NUMBER. Enter all project numbers as they appear in the report, e.g. 1F665702D1257; ILIR.

5e. TASK NUMBER. Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

5f. WORK UNIT NUMBER. Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

6. AUTHOR(S). Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, J, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES). Self-explanatory.

8. PERFORMING ORGANIZATION REPORT NUMBER. Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

9. SPONSOR/MONITOR'S ACRONYM(S) AND ADDRESS(ES). Enter the name and address of the organization(s) financially responsible for and monitoring the work.

10. SPONSOR/MONITOR'S REPORT NUMBER(S). Enter report number as assigned by the sponsoring/monitoring agency, if available, e.g. BRL-TR-829; -215.

11. DISTRIBUTION/AVAILABILITY STATEMENT. Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/ restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

12. SUPPLEMENTARY NOTES. Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

13. ABSTRACT. A brief (approximately 200 words) factual summary of the most significant information.

14. SUBJECT TERMS. Key words or phrases identifying major concepts in the report.

15. SECURITY CLASSIFICATION. Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

16. LIMITATION OF ABSTRACT. This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.
Go Ahead of Malware’s Infections and Controls: Towards New Techniques for Proactive Cyber Defense

Final Report
Grant #: FA9550-13-1-0077
Duration: 2/15/2013 – 8/14/2016
Program Manager: TRISTAN NGUYEN
PI: Guofei Gu
Texas A&M Engineering Experiment Station
November 10, 2016

Abstract

We are currently witnessing a sea change in cyber crimes carried out by malware, i.e., from fun/fame driven to profit driven. With strong economic incentives, cyber criminals are writing much better malware. Even worse, the attack-defense arms race actually favors attackers, because they are more proactive in designing/testing new attacks.

To address the challenge, this research aims at designing novel proactive techniques to defend against current and next generation malware/attacks. We have worked on three areas. First, we developed several new advanced malware analysis and detection techniques for both PC and emerging Android smartphone platforms. Second, we developed two novel proactive techniques for active, robust, fast and scalable malware server infrastructure detection at the Internet scale. In our evaluation, our prototype systems (CyberProbe and AutoProbe) can successfully uncover hundreds of malicious servers on the Internet in a few hours, many of them unknown to existing techniques/blacklists. Finally, for proactive defense, it is very important to
look forward to the security issues in the next generation networking technology, which is widely believed to be Software-Defined Networking (SDN). We have discovered several new vulnerabilities in SDN and developed new defense for SDN, as well as new security capabilities using SDN.

1 Introduction

Nowadays, most cyber attacks and fraudulent activities are carried out by malware, which includes viruses, trojans, worms, spyware, and recently botnets. One of the most significant changes in current cyber attacks is that unlike traditional attacks that are mainly launched for fun or fame, the new waves of cyber crimes are mainly to pursue profits, and this time the compromised computers become valuable resources (instead of merely targets) for making profits for cyber criminals. With such strong economic incentives, cyber criminals are writing much better malware programs that can evade existing widely used signature-based anti-virus tools. To make things even worse, the attack-defense arms race actually favors attackers, because they are more proactive in designing/testing new (cheap) attacks against one certain vulnerability.

This research aims at investigating new “game-changing” defense approaches, particularly designing novel techniques to defend against current and next generation malware in a proactive way. In particular, we focus on the following three areas:

- Advanced malware analysis and detection. To deal with advanced malware attacks targeting current and emerging platforms, we developed new advanced malware analysis and defense techniques such as AutoVac and GoldenEye for the PC platform, as well as DroidMiner, Dagger and UIPicker for the Android smartphone platform.

- Internet-scale malware server infrastructure detection. Malware server infrastructure is of extreme importance for modern malware operation, thus it is more fundamental and meaningful for defenders to detect these infrastructure then infected hosts. To this end, we have developed two novel proactive techniques for active, robust, fast and scalable malware server infrastructure detection at the Internet scale. In our evaluation, our prototype systems (CyberProbe and AutoProbe) can
successfully uncover hundreds of malicious servers on the Internet in a few hours, many of them unknown to existing techniques/blacklists.

- Securing Next Generation Networks: SDN Security. For proactive defense, it is very important to look forward to the security issues in the next generation networking technology, which is widely believed to be Software-Defined Networking (SDN). We have discovered several new vulnerabilities in SDN (such as topology poisoning attacks and data-to-control plan saturation attacks) and developed new defense for SDN (such as TopoGuard and FloodGuard), as well as new security capabilities using SDN (such as PBS).

2 Accomplishments in Area 1: Advanced Malware Analysis and Detection

2.1 AutoVac: Towards Automatically Extracting System Resource Constraints and Generating Vaccines for Malware Immunization

We design and implement AutoVac, a new system for automatically extracting the system resource constraints from malware code and generating vaccines, which can be then infected into other computers to be immune from future infections from the same malware or its polymorphic variants. We have evaluated AUTOVAC on a large set of real-world malware samples and successfully extracted working vaccines for many families including high-profile Concker, Sality and Zeus. We believe AUTOVAC represents an appealing technique to complement existing malware defenses. This work was presented at ICDCS’13 [7].

2.2 GoldenEye: Efficiently and Effectively Unveiling Malwares Targeted Environment

We design a new system GoldenEye to detect targeted malware. A critical challenge when combating malware threat is how to efficiently and effectively identify the targeted victims environment, given an unknown malware sample. Unfortunately, existing malware analysis techniques either use a limited, fixed set of analysis environments (not effective) or employ expensive,
time-consuming multi-path exploration (not efficient), making them not well-suited to solve this challenge. As such, we propose a new dynamic analysis scheme to deal with this problem by applying the concept of speculative execution in this new context. Specifically, by providing multiple dynamically created, parallel, and virtual environment spaces, we speculatively execute a malware sample and adaptively switch to the right environment during the analysis. Interestingly, while our approach appears to trade space for speed, we show that it can actually use less memory space and achieve much higher speed than existing schemes. We have implemented a prototype system and evaluated it with a large real-world malware dataset. The experimental results show that GoldenEye outperforms existing solutions and can effectively and efficiently expose malwares targeted environment, thereby speeding up the analysis in the critical battle against the emerging targeted malware threat. This work was presented at RAID’14 [8].

2.3 DroidMiner: Automated Mining and Characterization of Fine-grained Malicious Behaviors in Android Applications

We also design a new Android malware detection system called DroidMiner. A distinguishing advantage is that, unlike existing approaches that require human effort to extract signatures or patterns of malicious activities, DroidMiner can automatically learn/mine malicious patterns in terms of their malicious program logic and abstract them into threat modalities. Unlike many existing approaches that can only provide yes/no answer to whether a given app is malicious, DroidMiner can further provide further evidence as to why the app is considered to be malicious by including a concise description of identified malicious behaviors. This work was presented at a top computer security conference, ESORICS’14 [9].

2.4 Dagger: Using Provenance Patterns to Vet Sensitive Behaviors in Android Apps

We propose Dagger, a lightweight system to dynamically vet sensitive behaviors in Android apps. Dagger avoids costly instrumentation of virtual machines or modifications to the Android kernel. Instead, Dagger reconstructs the program semantics by tracking provenance relationships and observing
apps runtime interactions with the phone platform. More specifically, Dagger uses three types of low-level execution information at runtime: system calls, Android Binder transactions, and app process details. System call collection is performed via Strace, a low-latency utility for Linux and other Unix-like systems. Binder transactions are recorded by accessing Binder module logs via sysfs. App process details are extracted from the Android /proc file system. A data provenance graph is then built to record the interactions between the app and the phone system based on these three types of information. Dagger identifies behaviors by matching the provenance graph with the behavior graph patterns that are previously extracted from the internal working logic of the Android framework. We evaluate Dagger on both a set of over 1200 known malicious Android apps, and a second set of 1000 apps randomly selected from a corpus of over 18,000 Google Play apps. Our evaluation shows that Dagger can effectively vet sensitive behaviors in apps, especially for those using complex obfuscation techniques. We measured the overhead based on a representative benchmark app, and found that both the memory and CPU overhead are less than 10%. The runtime overhead is less than 63%, which is significantly lower than that of existing approaches. This paper was presented in Securecomm15 [10].

2.5 UIPicker: User-Input Privacy Identification in Mobile Applications

We design a new security solution to identify sensitive user inputs in smartphone apps for privacy protection. When it comes to today's program analysis systems, only those data that go through well-defined system APIs can be automatically labeled. In our research, we show that this conventional approach is far from adequate, as most sensitive inputs are actually entered by the user at an app's runtime: in our research, we inspect 17,425 top apps from Google Play, and find that 35.46% of them involve sensitive user inputs. Manually marking them involves a lot of effort, impeding a large-scale, automated analysis of apps for potential information leaks. To address this important issue, we present UIPicker, an adaptable framework for automatic identification of sensitive user inputs. UIPicker is designed to detect the semantic information within the application layout resources and program code, and further analyze it for the locations where security-critical information may show up. This approach can support a variety of existing security
analysis on mobile apps. We further develop a runtime protection mechanism on top of the technique, which helps the user make informed decisions when her sensitive data is about to leave the device in an unexpected way. We evaluate our approach over 200 randomly selected popular apps on Google-Play. UIPicker is able to accurately label sensitive user inputs most of the time, with 93.6% precision and 90.1% recall. This paper was presented in a top computer security conferee, USENIX Security15 [3].

3 Accomplishments in Area 2: Internet-scale Malware Server Infrastructure Detection

3.1 CyberProbe: Towards Internet-Scale Active Detection of Malicious Servers

We propose and design CyberProbe, a new active probing approach for detecting malicious servers and compromised hosts on Internet. Compared with existing defenses, our active probing approach is fast, cheap, easy to deploy, and achieves Internet scale. We have used CyberProbe to identify 151 malicious servers and 7,881 P2P bots through 24 localized and Internet-wide scans. Of those servers 75% are unknown to publicly available databases of malicious servers, indicating that CyberProbe can achieve up to 4 times better coverage than existing techniques. This paper was presented in a top computer security conferee, NDSS’14 [4].

3.2 AutoProbe: Towards Automatic Active Malicious Server Probing Using Dynamic Binary Analysis

We propose a new system (AutoProbe) to automatically generate effective and efficient network fingerprints of remote malicious servers through dynamic binary analysis. With our technique, we can automatically analyze malware binaries, extract network fingerprints (even if the remote servers are not alive), and then automatically scan the whole internet (or a designated subnet) to identify hidden malicious servers (e.g., C&C servers for botnets, or exploit servers, redirection servers and payment servers used in malware ecosystem). In our evaluation, AutoProbe can successfully uncover hundreds of malicious servers on the Internet, many of them unknown to...
existing blacklists. This paper was presented in a top computer security conference, ACM CCS’14 [6].

4 Accomplishment Area 3: SDN Security

4.1 Poisoning Network Visibility in Software-Defined Networks: New Attacks and Countermeasures

As part of our research on discovering new vulnerabilities ahead of attackers, we have identified a new attack vector against the emerging Software-Defined Network (SDN), which is widely considered as our future networking paradigm. We demonstrate that this new attack can effectively poison the network topology information, then further successfully launch hijacking, denial of service or man-in-the-middle attacks in SDN. According to our study, all current major SDN controllers in the market (e.g., Floodlight, OpenDaylight, Beacon, and POX) are affected, i.e., they are subject to the Network Topology Poisoning Attacks. To mitigate such attacks, we design TopoGuard, a new security extension to SDN controllers, that can provide automatic and real-time detection of Network Topology Poisoning Attacks. This work was presented in a top computer security conference, NDSS 2015 [2].

4.2 FloodGuard: A DoS Attack Prevention Extension in Software-Defined Networks

We design a new security solution for one serious SDN-specific attack, i.e., data-to-control plane saturation attack, which overloads the infrastructure of SDN networks. In this attack, an attacker can produce a large amount of table-miss packet in messages to consume resources in both control plane and data plane. To mitigate this security threat, we introduce an efficient, lightweight and protocol-independent defense framework for SDN networks. Our solution, called FloodGuard, contains two new techniques/modules: proactive flow rule analyzer and packet migration. To preserve network policy enforcement, proactive flow rule analyzer dynamically derives proactive flow rules by reasoning the runtime logic of the SDN/OpenFlow controller and its applications. To protect the controller from being overloaded, packet migration temporarily caches the flooding packets and submits them to the
OpenFlow controller using rate limit and round-robin scheduling. We evaluate FloodGuard through a prototype implementation tested in both software and hardware environments. The results show that FloodGuard is effective with adding only minor overhead into the entire SDN/OpenFlow infrastructure. This paper was presented in a top computer security conference, DSN15 [5].

### 4.3 PBS: Towards SDN-Defined Programmable BYOD Security

We design a new programmable network security solution for the emerging BYOD (bring your own device) paradigm. While existing solutions such as Mobile Device Management (MDM) focus mainly on controlling and protecting device data, they fall short in providing a holistic network protection system. New innovation is needed in providing administrators with sophisticated network policies and control capabilities over the devices and mobile applications (apps). In this work, we present PBS (Programmable BYOD Security), a new security solution to enable fine-grained, application-level network security programmability for the purpose of network management and policy enforcement on mobile apps and devices. Our work is motivated by another emerging and powerful concept, SDN (Software-Defined Networking). With a novel abstraction of mobile device elements (e.g., apps and network interfaces on the device) into conventional SDN network elements, PBS intends to provide network-wide, context-aware, app-specific policy enforcement at run-time without introducing much overhead on a resource-constrained mobile device, and without the actual deployment of SDN switches in enterprise networks. We implement a prototype system of PBS, with a controller component that runs a BYOD policy program on existing SDN controllers and a client component, PBS-DROID, for Android devices. Our evaluation shows that PBS is an effective and practical solution for BYOD security. This work was presented in a top computer security conference, NDSS 2016 [1].

### References


DISTRIBUTION A: Distribution approved for public release.
1. Report Type
   Final Report

Primary Contact Email
Contact email if there is a problem with the report.
   guofei@cse.tamu.edu

Primary Contact Phone Number
Contact phone number if there is a problem with the report
   9798452475

Organization / Institution name
   Texas A&M Engineering Experiment Station

Grant/Contract Title
   The full title of the funded effort.
   Go Ahead of Malware's Infections and Controls: Towards New Techniques for Proactive Cyber Defense

Grant/Contract Number
   AFOSR assigned control number. It must begin with "FA9550" or "F49620" or "FA2386".
   FA9550-13-1-0077

Principal Investigator Name
   The full name of the principal investigator on the grant or contract.
   Guofei Gu

Program Officer
   The AFOSR Program Officer currently assigned to the award
   TRISTAN NGUYEN

Reporting Period Start Date
   02/15/2013

Reporting Period End Date
   08/14/2016

Abstract
   We are currently witnessing a sea change in cyber crimes carried out by malware, i.e., from fun/fame driven to profit driven. With strong economic incentives, cyber criminals are writing much better malware. Even worse, the attack-defense arms race actually favors attackers, because they are more proactive in designing/testing new attacks.

To address the challenge, this research aims at designing novel proactive techniques to defend against current and next generation malware/attacks. We have worked on three areas. First, we developed several new advanced malware analysis and detection techniques for both PC and emerging Android smartphone platforms. Second, we developed two novel proactive techniques for active, robust, fast and scalable malware server infrastructure detection at the Internet scale. In our evaluation, our prototype systems (CyberProbe and AutoProbe) can successfully uncover hundreds of malicious servers on the Internet in a few hours, many of them unknown to existing techniques/blacklists. Finally, for proactive defense, it is very important to look forward to the security issues in the next generation networking technology, which is widely believed to be Software-Defined Networking (SDN). We have discovered several new

DISTRIBUTION A: Distribution approved for public release.
vulnerabilities in SDN and developed new defense for SDN, as well as new security capabilities using SDN.

Distribution Statement
This is block 12 on the SF298 form.

Distribution A - Approved for Public Release

Explanation for Distribution Statement
If this is not approved for public release, please provide a short explanation. E.g., contains proprietary information.

SF298 Form
Please attach your SF298 form. A blank SF298 can be found here. Please do not password protect or secure the PDF. The maximum file size for an SF298 is 50MB.

sf0298.pdf

Upload the Report Document. File must be a PDF. Please do not password protect or secure the PDF. The maximum file size for the Report Document is 50MB.

main.pdf

Upload a Report Document, if any. The maximum file size for the Report Document is 50MB.

Archival Publications (published) during reporting period:


DISTRIBUTION A: Distribution approved for public release.


New discoveries, inventions, or patent disclosures:
Do you have any discoveries, inventions, or patent disclosures to report for this period?
No

Please describe and include any notable dates
Do you plan to pursue a claim for personal or organizational intellectual property?
Changes in research objectives (if any):
Change in AFOSR Program Officer, if any:
Extensions granted or milestones slipped, if any:
1 six-month extension was granted.

AFOSR LRIR Number
LRIR Title
Reporting Period
Laboratory Task Manager
Program Officer
Research Objectives
Technical Summary

Funding Summary by Cost Category (by FY, $K)

<table>
<thead>
<tr>
<th></th>
<th>Starting FY</th>
<th>FY+1</th>
<th>FY+2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Salary</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Equipment/Facilities</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supplies</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Report Document
Report Document - Text Analysis
Report Document - Text Analysis
Appendix Documents

2. Thank You

E-mail user
Nov 18, 2016 13:44:19 Success: Email Sent to: guofei@cse.tamu.edu

DISTRIBUTION A: Distribution approved for public release.