REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
17-08-2015	Final Report	10-Nov-2009 - 9-Nov-2013

4. TITLE AND SUBTITLE

Final Report: Automatic Identification & Mitigation of Unauthorized Information Leaking from Enterprise Networks

5a. CONTRACT NUMBER

5b. GRANT NUMBER

W911NF-10-C-0018

5c. PROGRAM ELEMENT NUMBER

606055

6. AUTHORS

Dr. Anup Ghosh, Dr. Angelos Stavrou, Christopher Greamo

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES

Secure Command, LLC
4972 Marshall Crown Rd.

Centreville, VA 20120 -6425

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

57376-CS-ST2.1

12. DISTRIBUTION AVAILIBILITY STATEMENT

Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT

Malicious code such as spyware, adware, key loggers, Trojans, rootkits, botnets and other unauthorized software pose serious threats to the DoD enterprise as they may be used to collect information, provide access, respond to remote commands, and exfiltrate data. The goal of this project was to develop and evaluate novel mechanisms to classify and identify malicious software running in the enterprise by examining program network traffic and automatically generate the appropriate profiles of network behavior for each program, which we call application network behavior signatures. Where current approaches develop signatures of known attacks, our approach is to

15. SUBJECT TERMS

Malware, Network Intrusion Detection, Transparent Network Proxy

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Anup Ghosh
UU	UU	UU	UU		19b. TELEPHONE NUMBER 703-993-4776

Report Title

Final Report: Automatic Identification & Mitigation of Unauthorized Information Leaking from Enterprise Networks

ABSTRACT

Malicious code such as spyware, adware, key loggers, Trojans, rootkits, botnets and other unauthorized software pose serious threats to the DoD enterprise as they may be used to collect information, provide access, respond to remote commands, and exfiltrate data. The goal of this project was to develop and evaluate novel mechanisms to classify and identify malicious software running in the enterprise by examining program network traffic and automatically generate the appropriate profiles of network behavior for each program, which we call application network behavior signatures. Where current approaches develop signatures of known attacks, our approach is to validate all out-going network sessions based on their application network behavior signatures. Our approach is two pronged: (1) we passively examine the network characteristics of applications using a set of transparent proxies located on the network edges that use packet fingerprinting algorithms, and (2) in addition to pure passive monitoring, we are developing active content challenge approaches to verifying the authenticity of programs sending outbound data.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations

Number of Presentations: 0.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

Received Paper

TOTAL:

Number of Manuscripts:

Books

Received Book

TOTAL:

Received Book Chapter

TOTAL:

Patents Submitted

Patents Awarded

Awards

Graduate Students

NAME	PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Names of Post Doctorates

NAME	PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Names of Faculty Supported

NAME	PERCENT_SUPPORTED	National Academy Member
Dr. Angelos Stavrou	0.10	
FTE Equivalent:	**0.10**	
Total Number:	**1**	

Names of Under Graduate students supported

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: 0.00

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:..... 0.00

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):..... 0.00

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense 0.00

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields: 0.00

Names of Personnel receiving masters degrees

NAME
Total Number:

Names of personnel receiving PHDs

NAME
Total Number:

Names of other research staff

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Sub Contractors (DD882)

1 a. George Mason University

1 b. 4400 University Drive, MS 4C6

Fairfax VA 220304422

Sub Contractor Numbers (c):

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): Research into the network traffic and protocols used by malware.

Sub Contract Award Date (f-1): 2/21/10 12:00AM

Sub Contract Est Completion Date(f-2): 10/31/12 12:00AM

1 a. George Mason University

1 b. 4400 University Drive, MSN 4C6

Fairfax VA 220304422

Sub Contractor Numbers (c):

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): Research into the network traffic and protocols used by malware.

Sub Contract Award Date (f-1): 2/21/10 12:00AM

Sub Contract Est Completion Date(f-2): 10/31/12 12:00AM

Inventions (DD882)

Scientific Progress

See attachment

Technology Transfer

Automatic Identification & Mitigation of Unauthorized Information Leaking from Enterprise Networks

Final Progress Report

November 27, 2012

Period of Performance: 11 November 2009 – 10 November 2012

DETECTION | PREVENTION | INTELLIGENCE

3975 University Drive, Suite 460, Fairfax, Virginia 22030 | (703) 352-7680 | Invincea.com | @invincea

Problem Statement

Malicious code such as spyware, adware, key loggers, Trojans, rootkits, botnets and other unauthorized software pose serious threats to the DoD enterprise as they may be used to collect information, provide access, respond to remote commands, and exfiltrate data. The goal of this project was to develop and evaluate novel mechanisms to classify and identify malicious software running in the enterprise by examining program network traffic and automatically generate the appropriate profiles of network behavior for each program, which we call *application network behavior signatures*. Where current approaches develop signatures of known attacks, our approach is to validate all out-going network sessions based on their application network behavior signatures. Our approach is two pronged: (1) we passively examine the network characteristics of applications using a set of transparent proxies located on the network edges that use packet fingerprinting algorithms, and (2) in addition to pure passive monitoring, we are developing *active content challenge* approaches to verifying the authenticity of programs sending outbound data.

Summary of Results

The goal of this project was to develop and evaluate novel mechanisms to classify and identify malicious software running in the enterprise by examining program network traffic involved in the exfiltration of data or malware command and control. Where current approaches attempt to use signatures of known attacks and typically focus on analyzing *inbound* network traffic, the approach pursued here is to validate all out-going network sessions based on their application network behavior to identify communications associated with command and control to botmasters on the Internet and/or data exfilration. This approach is two pronged: (1) the system passively examine the network characteristics of applications using a set of transparent proxies located on the network edges that use protocol fingerprinting, and (2) in addition to pure passive monitoring, the proxy *creates active content challenges* call “program interactive proofs” or PIPs to software to differentiate legitimate from sophisticated malware. PIPs were inspired by the popular CAPTCHA (“Completely Automated Public Turing test to tell Computers and Humans Apart”) mechanism deployed by many websites to discern people from machines. In the case of PIPs, the system automatically generates challenge/responses that are specifically designed to distinguish legitimate software from malware attempting to mimic legitimate protocols in an attempt to blend in and avoid detection. This approach is non-disruptive to applications and transparent to users. The methodology we follow attempts to reveal the sophisticated malware by both classifying applications that are known to be good from applications that are either unknown or known to be malicious.

In this project, we developed a network-based proxy to automatically examine all network traffic emanating from the enterprise through Internet egress points. We successfully differentiated the most popular browsers (Internet Explorer, Firefox, Opera) from malicious software by passively analyzing their outbound traffic. Using PIPs, we were also able to detect sophisticated malware that imported browser components, such as Internet Explorer. We also extended our solution to work with other common enterprise network protocols such as VoIP protocols, including Session Initiation Protocol (SIP), Session Description Protocol (SDP), Real-time Transport Control Protocol (RTCP), and Real-time Transport Protocol (RTP). Our results showed that our passive

analysis approach does extend to VoIP protocols and successfully differentiated multiple VoIP implementations including Asterix, sipX, PBX, and Skype.

By categorically identifying all network traffic emanating to the Internet as known good, malicious or unknown, we were able to identify machines within an enterprise that are compromised and may be under the command and control of an outside entity.

This approach overcomes several pitfalls of the on-host approach:

1) The approach uses an agentless/clientless passive network solution that obviates the need for an enterprise-wide roll-out, considered by industry to be very expensive. Also, with an agentless solution, we don't have to worry about stepping on or being stepped on by other client-side software such as anti-virus software, for example DoD's Host-Based Security Solution (HBSS). We also do not have to worry about affecting the reliability of users' machines.

2) We do not have to update signatures on client machines enterprise wide. In fact our solution does not require signatures, though signatures enhance the identification of known malware. Any signature updates occur only to one machine – the network appliance. Our approach automatically creates signatures through an automated fingerprinting approach for network traffic.

3) The comprehensive approach to fingerprinting all network traffic emanating from the enterprise network will identify all known good, known bad, and unknown traffic. This changes signatures from a necessity to finding malware to merely additional information about the found netted malware.

4) Since this approach does not live on the host, it is not susceptible to being subverted or manipulated by rootkits or other on-host process-hiding evasion techniques. Rather, we detect the malware when it attempts to communicate the Internet preventing it from exfiltrating data, propagate, receive new targets or even update itself to the new version. All of the aforementioned activities can be detected and potentially further analyzed by moving the malware to execute on a controlled, clean from sensitive information environment.

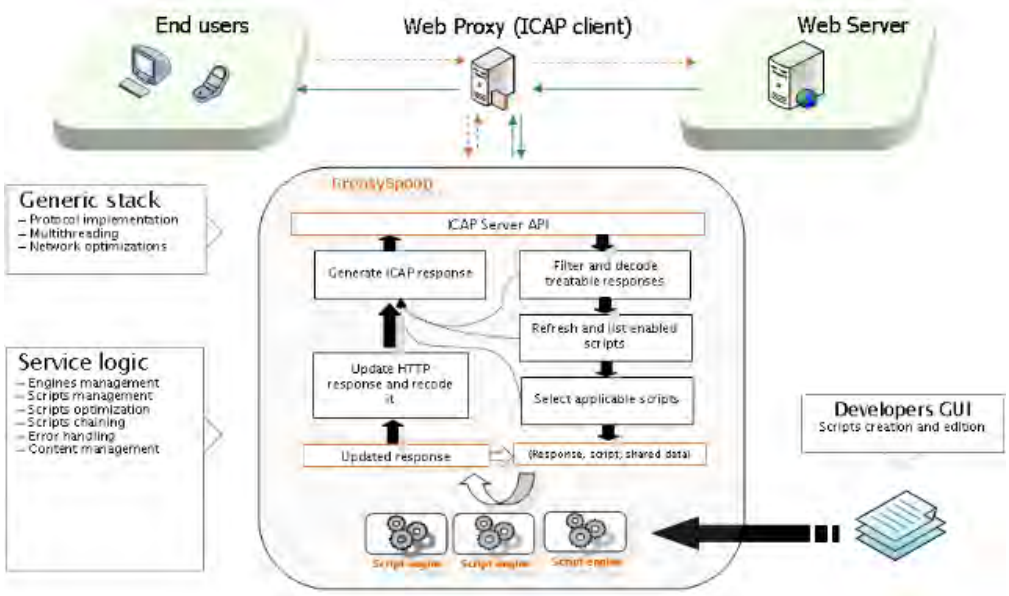

Figure 1: This figure depicts the overall system architecture. The ICAP compliant transparent proxy (prototyped using the opensource GreasySpoon project) is mediating all traffic both encrypted and non-encrypted when a client initiates communications with a server outside of the enterprise. No communications can pass without being analyzed by the transparent proxy.

Using passive analysis, the system characterizes all network traffic. Almost all *prior art* has focused on detecting malicious software or traffic. Approaches have included using known signatures, generating signatures from repetitive features that would be present in worm-based attacks to anomalous flow detection. Since this approach characterizes *all* network traffic, we are not actually "searching" for malicious traffic. Rather, we classify network traffic by its originating program based on its fingerprint using network level analysis including header information such as timing and source/destination addresses and ports. Moreover, we harness the HTTP headers' ordering and special HTTP headers to identify known browser applications and then decide which bucket that program falls in: traffic from already encountered program, known netted malware, or unknown.

The HTTP passive analysis uses two types of features – (1) the fields present in the HTTP header and (2) the order of the fields. Figures 2 and 3 below show examples of each feature type and how they map to browser type. These features are then provided to a classifier to identify the browser type. The classifier that we chose was a Decision Tree approached based on ID3. Unknown browsers are flagged as potential malware. For known popular browsers, the identification is then used to drive selection of an appropriate active challenge to validate the browser and differentiate it from malware mimicking the browser's HTTP header.

Host	User_agent	Accept	Language	Encoding	Charset	Keep-Alive	Connection	UA-CPU	Type
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Firefox
Y	Y	Y	Y	NULL	Y	NULL	Y	NULL	Malware
Y	Y	Y	Y	Y	NULL	Y	Y	Y	IE6
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Firefox
Y	Y	Y	Y	Y	Y	NULL	NULL	NULL	Malware
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Opera
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Opera
Y	Y	Y	Y	Y	NULL	NULL	Y	Y	IE6
Y	Y	Y	NULL	Y	NULL	Y	Y	Y	IE7
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Opera
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Firefox
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Opera
Y	Y	Y	NULL	NULL	NULL	Y	Y	NULL	Malware
Y	Y	Y	NULL	Y	NULL	Y	Y	Y	IE7
Y	Y	Y	Y	Y	Y	Y	Y	NULL	Firefox
Y	Y	Y	Y	Y	NULL	NULL	NULL	NULL	IE6
Y	Y	Y	NULL	Y	NULL	Y	Y	Y	IE7

Figure 2: Example HTTP header features mapped to Browser type

	Host	User-Agent	Accept	Accept-Language	Accept-Encoding	Accept-Charset
Firefox	0	1	2	3	4	5
IE7	4	3	0	Selected	2	
IE6	4	3	0	1	2	
Opera	1	0	2	3	5	4

	Keep-Alive	Connection	UA-CPU	Referer	If-Modified-Since
Firefox	6	7			
IE7		5	1	Selected	Selected
IE6		5		Selected	
Opera					

Figure 3: Example HTTP header ordering features mapped to Browser type

In addition to passive analysis for HTTP, we performed a study of the existing Voice over IP clients and Servers and their legitimate protocol behavior and characteristics. To that end, we analyzed passively the network behavior of the following clients and we were able to identify them by their order of headers and other communication characteristics. Table 1, has a summary of the features that we used to detect popular VoIP clients based on the order used in the network protocol headers:

SIP Client	Element Order	Patterns
Ekiga	Sent-by port, branch, rport	[ip]:[port];branch=<z...>;rport
linphone	Sent-by port, rport, branch	[ip]:[port];rport;branch=<z...>
SJPhone	branch, port	[ip];branch=<z...>;rport
X-ten Lite	Sent-by port, branch, rport	[ip]:[port];branch=<z...z->;rport

Table1: Classification of different VoIP clients based on the header element order observed on their network communications.

Furthermore, we identified variations in the way that clients implement the "FROM" and "TO" fields characterizing some of the clients:

SIP Client	Element Order	Patterns
linphone	SIP from address, SIP tag	From:<sip[username]@[ip]>;tag=...
All others	SIP Display Info, SIP from address, SIP tag	From: "<display_name>"sip[username]@[ip]>;tag=...
X-ten Lite	SIP from address, SIP tag	From:<sip[username]@[ip]>;tag=...
All others	SIP Display Info, SIP from address, SIP tag	From: "<display_name>"sip[username]@[ip]>;tag=...

Table2: Classification of different VoIP clients based on the "FROM" and "TO" fields observed on their network communications.

Our experiments showed that using passive detection, the system can correctly detect 36.2% of the representative malware that we randomly sampled from both in an internal premises honeypot and from a Google provided malware feed. Figure 2 depicts the port distribution of the malware test data set. Figure 3 illustrates and compares the performance of systems with passive analysis and active challenge techniques. Notice that the range of the ports that the malware attempts to exploit has shifted towards HTTP and HTTPS traffic from pure IRC traffic that was prevalent couple of years ago. Furthermore, it is clear from Figure 3 that the passive analysis, although still effective, is not sufficient alone. Indeed, even if we collect a diverse set of data and applications to train our network sensor, purely passive network analysis is unable to recognize malware that "mimics" good/known web browser protocol behavior.

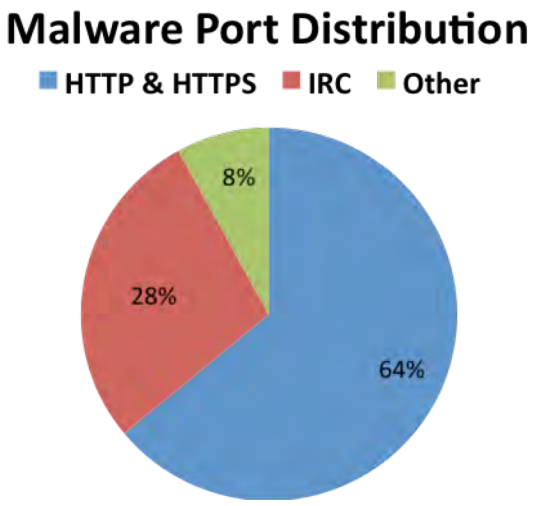

Figure 2: Malware port distribution for tested malware (425 instances). The majority of the command and control channels are HTTP and HTTPs.

To address the limits of the passive techniques alone and to ferret out malware that mimics the traffic behavior of legitimate applications, we introduced a novel, active detection mechanism, which we call "Program Interactive Proofs" or PIPs.

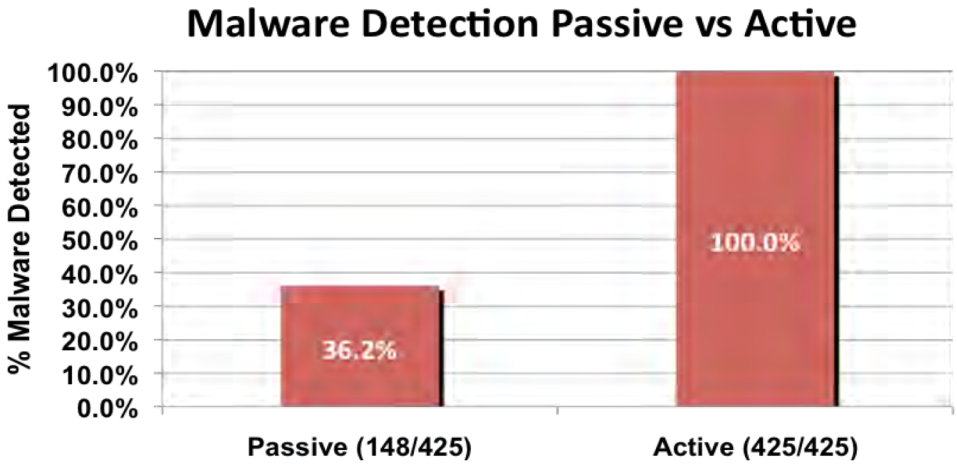

Figure 3: Detected malware for 425 instances samples from our malware corpus. The passive analysis techniques alone are insufficient to detect the majority of malware. On the other hand, the active analysis techniques have 100% detection rate with the tested malware corpus.

Unlike a traditional CAPTCHA that requires users be involved in the identification process by solving a puzzle, our approach requires no user involvement or application modification. Instead, we offer a seamless user experience with virtually no delays in most of the cases. The overall

mechanism is presented in Figure 4. First, the Browser attempts to communicate and fetch a page from the network. The transparent proxy generates an active content challenge using a unique, random hash in response to this request. The active content challenge validates the inherent Browser capabilities by making use of one or more of the HTML, Javascript, Flash, and graphics rendering engines. The challenge is an encoded redirect request, which is triggered only after the page is processed by the targeted browser engine(s). By making use of the browsers rendering engine, we can effectively distinguish between malicious and benign network traffic. The injected code is not visible to the users at all since the generated page has no content to display but rather the redirection request. Only if the client successfully replies to that request he is allowed to receive his initial fetch request. Notice, that the end-user is not involved in the process. Malware is revealed because it is unable to generate a valid response to the active content challenge but rather attempts to re-connect to the same or another, alternate website unsuccessfully. Therefore, the core of our approach is to frustrate the communication of the malware by injecting traffic that the malware is incapable to parse and generate a valid response contrary to the legitimate browser or legitimate application, in general.

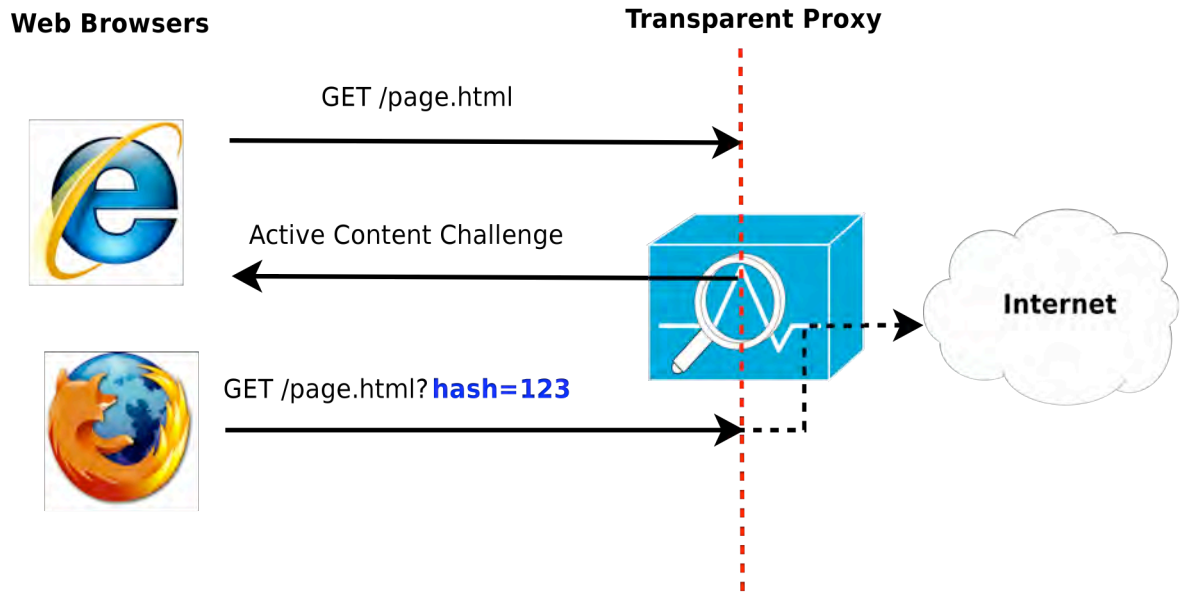

Figure 4: The browser attempts to fetch a page from the network. Upon reception of the http request, the proxy generates an active content challenge or PIP, which is seamlessly and transparently returned to the client software. The PIP is designed to perform a calculation that only a fully implemented, legitimate browser would be able to correctly perform. Malware is identified by either not responding at all (as the PIP tends to break the fragile malware software stack) or by returning the wrong answer. The correct challenge answer, as provided by a legitimate browser, results in the original request being forwarded to the Internet server and the resulting response (an HTML page in this case), being returned to the client.

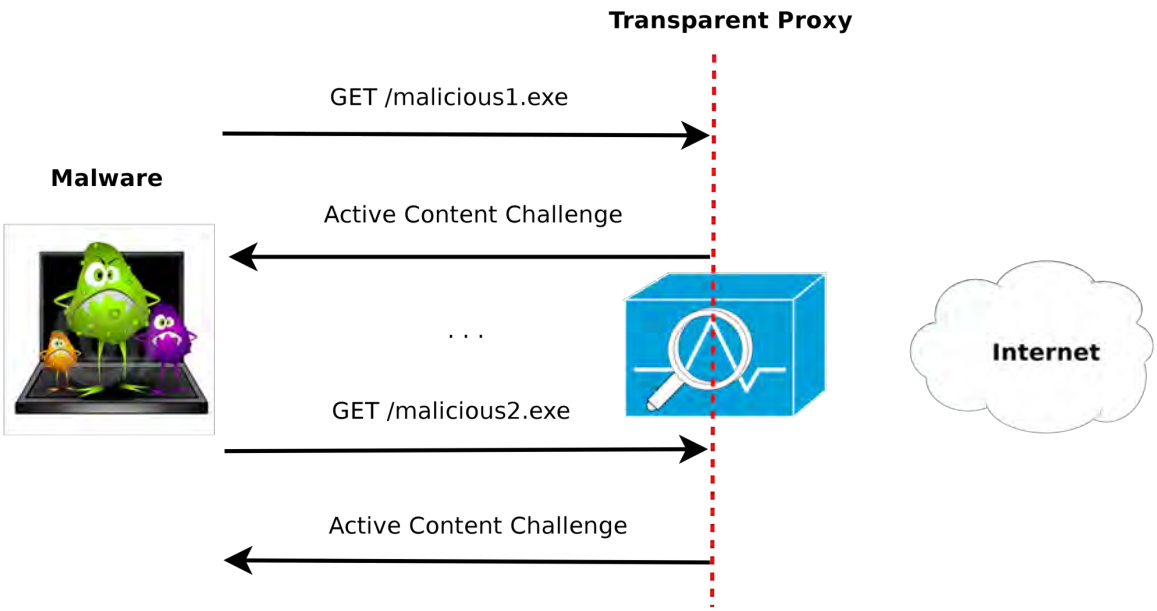

Figure 5:This figure shows the interaction between malware (a downloader in this case) and the transparent proxy. The malware attempts to use HTTP to download additional stages of the attack but all requests are met with an active content challenge or PIP and the malware is unable to correctly respond. The result is that no requests are forwarded by the proxy to the Internet and no further malware is downloaded. Additionally, the proxy is able to use the non-response from the client as an indication of compromise and alert system administrators to the infection.

In conclusion, the project detailed in this final report explored a combination of passive and active challenge techniques for identifying malware operating in enterprise networks based solely on outbound network traffic and not requiring agent or host level monitoring. We demonstrated that while passive techniques are effective at detecting some malware, it is necessary to combine both passive and active techniques in order to achieve full coverage (based on the malware corpus used in this study). We developed and demonstrated a novel active challenge technique called “Program Interactive Proofs” or PIPs that require no participation from the user and with minimal impact to the user’s experience. Finally, we presented and demonstrated a deployable architecture and end-to-end system prototype based on a transparent ICAP compliant network proxy which could be deployed at Internet access points from an enterprise.