MAPPER: A Mobile Application Personal Policy Enforcement Router for Enterprise Networks

Original

Availability:
This version is available at: 11583/2560943 since:

Publisher:
IEEE

Published
DOI:10.1109/EWSDN.2014.9

Terms of use:
openAccess
This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)
**Abstract**—MAPPER is a system for enforcing user-specific policies based on the availability of access nodes that support the capability to dynamically load and execute processing modules on the data path. This work leverages a network access node that, after authenticating a connecting user, loads a set of lightweight virtual machines that process traffic terminated on the user device to implement articulated user-specific access policies. Specifically, we demonstrate how a man-in-the-middle-proxy module, dynamically and opportunistically combined with a module capable of mobile application identification, can implement complex access policies. The man-in-the-middle-proxy module enables MAPPER policies to be applied to both clear and HTTPS traffic, while an intelligent traffic classification system, provides support for policies based on over 250,000 mobile apps spanning both Android and iOS platforms.

**Keywords**—network; policy; virtualization; apps;

**I. INTRODUCTION AND MOTIVATION**

MAPPER (Mobile Application Personal Policy Enforcement Router) is a network access point that, upon authenticating a connecting user, loads a set of modules to process network traffic to/from the user’s device, and implements user-specific access policies based not only on content but also on the user applications that generate the traffic. This is in contrast to state-of-the-art firewall systems deployed within enterprise networks that rely largely on IP/domain-name/port based policy formulation and enforcement. Such conventional approaches are increasingly becoming untenable due to two primary reasons: (a) acceptance of user-owned devices (most smartphones and tablets) within enterprise networks (a.k.a. the bring-your-own-device to work phenomenon), and (b) the unprecedented proliferation of mobile applications (roughly 1.75 million for Android and iOS at last count). This poses new challenges on several fronts. First, a significant portion of the mobile applications use HTTP and HTTPS as the transport layer protocol, and are thus indistinguishable from the conventional web-traffic. The advent of Web 2.0, which facilitates rapid development of web-based and distributed applications, has only accentuated the problem. Second, while some applications (e.g. mail clients, calendars), may actually be useful to employees for their daily work and are hence a necessity; others, not equally benign, may compromise network and information security. Finally, the same application developed for different platforms (e.g. Android vs. iOS) or user-devices (e.g. Samsung Galaxy vs. HTC One) may have different security vulnerabilities. Therefore, for high resolution fine-grained traffic monitoring, provisions must be made to segregate application level traffic, originating from different user-devices and across platforms, effectively.

Next, and perhaps more importantly, there is the question of roles and privileges of individuals within an enterprise: not all employees are equals. While certain employees might need to access and/or share sensitive information with prospective clients for business objectives, others should not be permitted to do so. As mobile applications increasingly become multi-faceted and complex, such exfiltration risks have only escalated. For example, popular services such as Facebook, now function as authentication gateways and substrates for a large ecosystem of applications (e.g. FarmVille). This can provide indirect access to sensitive information to undesired, and potentially malicious, third parties. In view of these challenges, it is imperative that modern policy formulation and enforcement frameworks/systems have the capability to identify and differentiate traffic generated by different applications, across platforms and devices, and impose policies based on user roles, to guarantee network and information security.

MAPPER enables network administrators to formulate and enforce user/role specific policies within the enterprise network, at application as well as content level granularity, without requiring access to an end-user’s device. User traffic segregation, for privacy and protection, is provided by allocating a user specific virtual machine, at the time of login, on a FROG (Flexible & pROGammable) mobile device [1], [2]. The user specific virtual machine runs data plane applications (a.k.a. net apps) that process, monitor, and filter the traffic associated with an individual user. MAPPER ensures that these net apps, as well as the policies they implement, persist seamlessly across network access points as the user moves within the enterprise. Similar migration is handled across multiple devices used by the same individual. A stand-alone module, interacting with the MAPPER system through a specific net app, handles mobile application identification and categorization [3]–[5]. Furthermore, by executing a man-in-the-middle proxy module [6], MAPPER also provides visibility into encrypted application traffic. Last but not least, MAPPER comes equipped with a simple yet flexible user-interface that enables a network administrator to define both the net apps and the corresponding policies for each user, cognizant of user-roles, applications, content, platforms and devices, in any combination.
Man-in-the-Middle Proxy (MiMP)

Mobile Application Identification Module

net apps and policies to migrate also across devices.

together with the user. Since this is true even when

device, thereby providing the experience of them moving

the event of an anomaly. Finally, when a user authenticates

also simplifies the management process and forensics in

privacy and protection of user data. Needless to say, it

ket, interest, and network function categories. This enables

labels for each identified flow. The labels include app mar-

Moreover, this module provides device and platform infor-

mobile applications across the iOS and Android platforms.

and classifying network flows generated by over 250K

with a different FROG device, MAPPER loads the same

the time of their connection to the wireless LAN. Upon

success, user profiles are loaded, i.e., net apps are down-

loaded from a net app marketplace, run and configured

with the policies defined for the users, FROG allocates

an exclusive virtual machine (VM) for each individual

user to runs a chain of network apps (e.g. firewalls,

network monitors etc). The network apps operate on every

packet generated during a user’s session. An obvious, and

desirable, side-effect of such user level segregation is the

privacy and protection of user data. Needless to say, it

also simplifies the management process and forensics in

the event of an anomaly. Finally, when a user authenticates

with a different FROG device, MAPPER loads the same

net app chain and policy configuration into the new FROG

device, thereby providing the experience of them moving

together with the user. Since this is true even when the

user connects with a difference device, we can consider

net apps and policies to migrate also across devices.

Mobile Application Identification Module [3]: A power-

ful application identification module capable of extracting

and classifying network flows generated by over 250K

mobile applications across the iOS and Android platforms.

Moreover, this module provides device and platform infor-

mation as well as application context in terms of category

labels for each identified flow. The labels include app mar-

kets, interest, and network function categories. This enables

policy formulation at different granularity levels, whereby

the network manager can white/blacklist individual mobile

applications, devices, platforms or even broad categories.

Man-in-the-Middle Proxy (MiMP) [6]: A solution to

provide visibility into encrypted application traffic by

terminating HTTPS sessions from the clients within the

wireless network and splicing them into HTTPS sessions
to external hosts. Upon the first connection of a device,

the user is required to install a certificate used for signing

server certificates required by TLS to implement HTTPS.

Although such practice might not be acceptable in gen-

eral, we consider it acceptable in an enterprise scenario

where MAPPER is operated under the supervision of the

corporate network administrator.

Content Filtering: As an added dimension to policy
definition, MAPPER enables keyword blacklisting in ap-
pllication content by means of a net app included in the
profile of a user. This can be used in conjunction with
application level policies, whereby a flow is considered
inadmissible if it contains a blacklisted term, even if the
application is allowed.

Policy Engine: By properly chaining net apps, MAP-
PER implements a simple yet modular policy engine that
enables articulation of enforceable policies based on
users, mobile applications, content, devices, and platforms,
or any arbitrary combination thereof. An easy-to-use UI
facilitates on-the-fly definition and modification of policies
by the administrator. A full net app chain will include (i)
 a MiMP Bridge net app to divert traffic to the MiMP module
and then insert it back in the net app chain, (ii) a Mobile
Application Filter net app to send relevant parts of the
traffic to the Mobile Application Identification Module,
receive back information on the mobile app generating
the traffic, and then verify its compliance with the user-
specific policy, and (iii) a Content Filter net app to check
compliance to the content policy.

ACKNOWLEDGEMENT

We would like to thank Ramya Gadiyaram for her help
and support with the setup of the demonstration system.

REFERENCES

[1] F. Risso and I. Cerrato, “Customizing data-plane processing in
dge routers,” in Software Defined Networking (EWSDN),

from the host to the network: Experiences, challenges,
and findings,” in Mobile Cloud Networking (MCN), 2013
Workshop on. IEEE, 2013.

“Networkprofiler: Towards automatic fingerprinting of an-

M. Baldi, and A. Nucci, “FLOWR: A Self-Learning System
for Classifying Mobile Application Traffic,” in Proceedings
of ACM SIGMETRICS, 2014.

ing mobile app usage patterns using in-app advertisements,”

http://mitmproxy.org/.