

Protecting E-healthcare Client Devices against Malware and Physical Theft

Daisuke Mashima, Abhinav Srivastava, Jonathon Giffin, and Mustaque Ahamad
School of Computer Science, Georgia Institute of Technology
{*mashima, abhinav, giffin, mustaq*}@cc.gatech.edu

Abstract

The growing adoption of electronic medical records will require that healthcare professionals and patients are able to access health information on devices such as laptops, personal computers, and even smart phones. Due to the sensitive nature of such information, it is important to secure client-side devices used in electronic healthcare systems that handle sensitive medical data. These devices can be targets of several kinds of attacks, including malware-based attacks or physical theft of devices. We present an approach and system architecture to secure client devices against such attacks.

1 Introduction

Medical records, which have been paper-based documents created and stored by healthcare providers, are now moving to electronic form to enable ubiquitous accessibility and effective sharing. To facilitate “meaningful use” of electronic health records (EHR), health information should be available to healthcare professionals and patients on devices where it can be conveniently accessed. For example, a small medical office may access it on their office computers, and a patient may access her information on her laptop or smart phone. Because of the sensitivity and value of health information, we can expect that attacks targeting health information will emerge. We focus on attacks that will target client devices where health information is accessed. Such attacks can roughly be categorized into two groups: (1) attacks using malware, such as key loggers, bots, etc., and (2) attacks resulting from physical theft of client devices.

If malware can successfully be installed on a client device, it could steal identity credentials, misuse stolen credentials to abuse systems, and disclose sensitive data to unauthorized parties. These attacks are possible even in the presence of anti-virus software due to their inability to handle zero-day attacks effectively. Moreover, other

security software, such as a host-level firewall, could be disabled when the devices are compromised. In the case of physical device theft, an adversary has full control over a device, and we can no longer assume any security mechanism deployed on the device remains functional. Furthermore, the attacker can misuse stored identity credentials to access healthcare systems, impersonating a legitimate client.

In this paper, we outline a novel electronic healthcare system architecture and security design for client devices to counter the threats of malware and theft.

2 Architecture Overview

Figure 1 shows the overview of our architecture. We assume a typical electronic healthcare system architecture, where a medical professional (i.e. a doctor or an EMT) or patient uses her device (e.g., a desktop/laptop PC) to access a network-attached repository or server that stores medical data, and augment it by introducing a networked monitoring system and authority. The monitoring system is run by an organization managing the client devices or a trusted third party. The authority, whose responsibility is explained later, is typically a privileged person or service in an organization with whom a client is affiliated.

Within a device, we create a trusted domain (trusted VM) that is isolated from an untrusted user domain (user VM) used by a human. We achieve this isolation via virtualization technologies available for commodity operating systems. Isolation among VMs prevents malware in a user VM from compromising code and data in a trusted VM. Therefore, we can establish a secure storage and execution environment in the trusted VM. Although in this paper we consider regular PCs as client devices, our scheme can be used for other types of clients, including smart phones, as long as a trusted domain can be established. For example, the Xen-ARM project (<http://wiki.xensource.com/xenwiki/XenARM>) is porting the Xen hypervisor to mobile

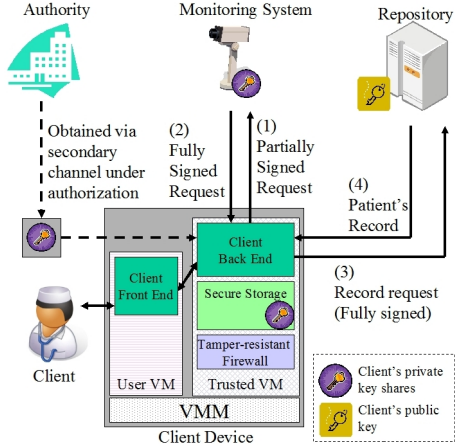


Figure 1: Overview of the system architecture

phones.

We split client-side software into two parts. Its security relevant functionality, such as cryptographic operations, management of credentials, and protocol handling, is implemented in the trusted VM (Client Back End or CBE) while application specific functionality that supports user interaction is in the user VM (Client Front End or CFE). Thus, CFE can be simply a viewer program or a regular web browser.

To ensure security, we require that a request made for health information must be signed by a client. A signature can be generated using a client's private key stored on her device. However, to deal with the attacks outlined above, we cannot store the full private key on the device and instead utilize a threshold signature based scheme [1], under which a subset of key shares are needed to create a valid signature. Such a scheme also allows us to ensure that a networked monitoring system cannot be bypassed by a malicious entity that has control of a client device.

More concretely, in our scheme, the client's private key is split into 3 shares and we use the 2-3 threshold signature scheme. One of the key shares is stored with the trusted VM on the client device, another one resides at the online monitoring system, and the third one is held by the authority. The monitoring system logs all requests issued by clients when signing them and runs an anomaly detector to detect identity theft attacks. A key share assigned to the authority is made available to clients only under special review and authorization, such as break-the-glass access in emergency cases.

Briefly, access to health information is supported as follows. A client initiates a request by using CFE, which sends a request to CBE. CBE then creates a request message, partially signs it with its key share, and sends it to the monitoring system. The monitoring system uses its key share to make another partial signature and combines two partial signatures into a full signature. After the fully

signed message is returned, CBE sends it to the repository, which then verifies the signature with the client's public key. If the verification is successful, then the requested data is returned to the CBE and CFE. Note that if the authority's key share can be obtained, then the CBE will be able to create a full signature without involving the monitoring system.

Although this does not appear in the protocol sketched here, we also propose to use a tamper-proof host-level firewall feature designed in [2] in the trusted VM to prevent malware in the user VM from sending data out of the client device. Since the user VM's access to the network interface of the device is mediated by the trusted VM or the virtual machine monitor (VMM in the figure), as long as they are secure, this scheme effectively works.

3 Security Analysis

Our architecture prevents malware-related attacks by establishing a trusted domain on a client device. Even if malware successfully installs itself in a user VM, isolation among VMs does not allow it to compromise credentials, including a key share, or CBE's functionality. Malware could attempt to send data in a user VM, such as downloaded medical data, to an attacker via the network, but the firewall in a trusted VM can block it.

In case a client device is stolen, even a trusted VM can be compromised. However, the 2-3 threshold signature scheme is still effective in detecting such attacks. Since the trusted VM has only one key share, it must either contact the monitoring system or obtain authorization from the authority to issue valid requests. In either case, suspicious access can be detected using techniques such as anomaly detection. Once detected or reported, the device can be immediately disabled by deleting the corresponding key shares on the monitoring system and authority.

4 Future Work

We are currently incorporating the design discussed in this paper into the MedVault system [3]. Based on this proof of concept prototype, we will conduct more detailed evaluation including extensive threat analysis.

References

- [1] Victor Shoup. "Practical Threshold Signatures," in Proc. of EUROCRYPTO 2000, 2000.
- [2] Abhinav Srivastava and Jonathon Giffin. "Tamper-resistant, Application-aware Blocking of Malicious Network Connections," in Proc. of RAID 2008.
- [3] MedVault, at <http://medvault.gtisc.gatech.edu/>.