CSC648: (due on Saturday 25th at 9 a.m)

Explain in detail **the classical authentication protocol(s) in your research area** and map it(them) if possible to what we have covered in the class **including the possible attacks**. For example, public and private Blockchains have different authentication protocols. You can cite different resources, but you have to have at least 2 research papers. I don't expect you to come up with new vulnerabilities in the protocols; instead I expect you to read and cite the research papers that investigated those issues if they are available.

The document is available on: https://github.com/EnzeXu/Computer_Security_Assignment2_CS648

Project Name	Computer Security - Assignment2 - Additional Assignment for CS648
Author	Enze Xu
Due	09/25/2021 at 9 a.m

Catalog

1 Authentication Protocols in IoT

Today, IoT applications span almost all sectors from medical to home automation and many more, carrying critical and sensitive data. There are different IoT device authentication methods, and each authentication method has its corresponding authentication protocols.

1.1 Why is authentication so important in IoT

[1] Surprisingly little attention has been paid to access-control-policy specification (expressing which particular users, in which contexts, are permitted to access a resource) or authentication (verifying that users are who they claim to be) in the home IoT. This state of affairs is troubling because the characteristics that make the IoT distinct from prior computing domains necessitate a rethinking of access control and authentication. Traditional devices like computers, phones, tablets, and smartwatches are generally used by only a single person. Therefore, once a user authenticates to their own device, minimal further access control is needed. These devices have screens and keyboards, so the process of authentication often involves passwords, PINs, fingerprint biometrics, or similar approaches.

Home IoT devices are fundamentally different. First, numerous users interact with a single home IoT device, such as a household's shared voice assistant or Internet-connected door lock. Widely deployed techniques for specifying access-control policies and authenticating users fall short when multiple users share a device. Complicating matters, users in a household often have complex social relationships with each other, changing the threat model. For example, mischievous children, parents curious about what their teenagers are doing, and abusive romantic partners are all localized threats amplified in-home IoT environments.

Furthermore, few IoT devices have screens or keyboards, so users cannot just type a password. While users could possibly use their phone as a central authentication mechanism, this would lose IoT devices' handsfree convenience, while naive solutions like speaking a password to a voice assistant are often insecure.

Moreover, IoT devices are generally designed to be resource-constrained with fit-for-computing and limited storage capabilities. Therefore, IoT devices are more prone to security attacks as most of these devices lack appropriate countermeasures.

IoT device authentication has to be different and considerably lightweight compared to an existing user or personal authentication method that is not directly applicable to IoT devices with limited resources. As a result, choosing the proper authentication method is of utmost importance to ensure robust security for IoT devices.

1.2 IoT device authentication methods

Single or one-factor authentication is the most basic form of IoT device authentication in which devices or users present something they know to verify their identity. Usernames and passwords are the most popular form of one-factor authentication.

Shared secrets like usernames and passwords are usually recycled at multiple places and are susceptible to various attacks. However, two-factor authentication extends the one-factor authentication of username/passwords by adding another layer where users or devices need to verify something they possess. The authentication extension could be a one-time password or something unique like fingerprints.

Three-factor or Multi-factor authentication extends security to the next level by combining multiple mechanisms to authenticate:

1) what you know (e.g., a password)

2) what you have (e.g., a phone, an email, a one-time password generator)

3) what you are (e.g., fingerprint, iris recognition, SSN)

1.3 IoT device authentication protocols

1.3.1 Protocols

1) TLS

Transport Layer Security (TLS), the successor of the now-deprecated Secure Sockets Layer (SSL), is a cryptographic protocol designed to provide communications security over a computer network. The protocol is widely used in applications such as email, instant messaging, and voice over IP, but its use as the Security layer in HTTPS remains the most publicly visible.

2) HTTPS

The principal motivations for HTTPS are authentication of the accessed website and protection of the privacy and integrity of the exchanged data while in transit. It protects against man-in-the-middle attacks, and the bidirectional encryption of communications between a client and server protects the communications against eavesdropping and tampering. The authentication aspect of HTTPS requires a trusted third party to sign server-side digital certificates.

3) SSH

Secure Shell (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network. Typical applications include remote command-line, login, and remote command execution, but any network service can be secured with SSH.

4) IMAP

In computing, the Internet Message Access Protocol (IMAP) is an Internet standard protocol used by email clients to retrieve email messages from a mail server over a TCP/IP connection.[1] IMAP is defined by RFC 3501

5) 3-D Secure

3-D Secure is a protocol designed to be an additional security layer for online credit and debit card transactions. The name refers to the "three domains" which interact using the protocol: the merchant/acquirer domain, the issuer domain, and the interoperability domain.

6) RFID

Radio-frequency identification (RFID) uses electromagnetic fields to automatically identify and track tags attached to objects. An RFID system consists of a tiny radio transponder, a radio receiver and transmitter. When triggered by an electromagnetic interrogation pulse from a nearby

RFID reader device, the tag transmits digital data, usually an identifying inventory number, back to the reader. This number can be used to track inventory goods.

7) BioAPI

BioAPI (Biometric Application Programming Interface) is a key part of the International Standards that support systems that perform biometric enrollment and verification (or identification). It defines interfaces between modules that enable software from multiple vendors to be integrated together to provide a biometrics application within a system, or between one or more systems using a defined Biometric Interworking Protocol (BIP).

1.3.2 Protocol table

I designed a table to classify authentication methods and the protocols they involve.

Group	Method	Examples	General Protocols	Specific Protocols
What you know	Password	Alexa's administrative password	1) TLS 2) HTTPS 3) SSH	/
What you have	Phone	Phone verification code (voice message) used to change your wifi password		/
	Email	Email verification code before you log into your Amazon account by IoT devices		4) IMAP
	Credit card	Use the bound credit card to reset your password		5) 3-D Secure
	Chip card	Open the electric door of your laboratory		6) RFID
What you are	Fingerprint	Smart fridge / Smart car		7) BioAPI
	Iris recognition	Smart door entrance		7) BioAPI
	Facial recognition	Smart door entrance		7) BioAPI

2 Possible Attacks against Protocols

Here are some vulnerabilities in some of the protocols I find in research papers. I summarize and cite them briefly.

2.1 Man-in-the-middle attack to the HTTPS protocol

[2] The man-in-the-middle (MITM) attack exploits the fact that the HTTPS server sends a certificate with its public key to the Web browser. If this certificate isn't trustworthy, the entire communication path is vulnerable. Such an attack replaces the original certificate authenticating the HTTPS server with a modified certificate. The attack is successful if the user neglects to double-check the certificate when the browser sends a warning notification. This occurs all too often—especially among users who frequently encounter selfsigned certificates when accessing intranet sites.

2.2 Multiple attacks to the RFID protocol

[3] RFID systems are vulnerable to a broad range of malicious attacks ranging from passive eavesdropping to active interference. Unlike in wired networks, where computing systems typically have both centralized and host-based defenses (e.g. firewalls), attacks against RFID networks can target decentralized parts of the system infrastructure since RFID readers and RFID tags operate in an inherently unstable and potentially noisy environment. Additionally, RFID technology is evolving quickly – the tags are multiplying and shrinking - and so the threats they are susceptible to, are similarly evolving. Thus, it becomes increasingly difficult to have a global view of the problem.

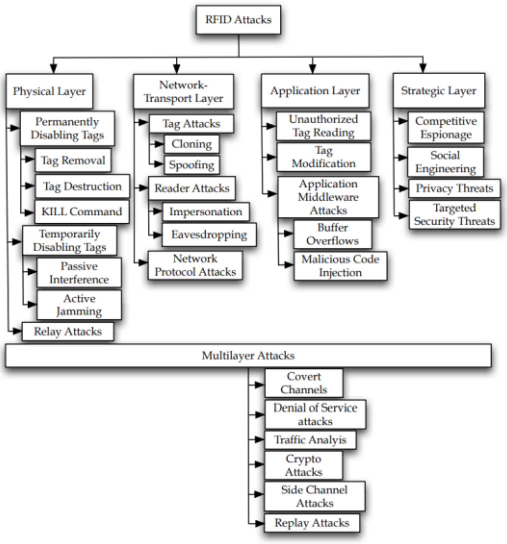

Fig. 2. Classification of RFID attacks.

2.3 Password recovery attack to the TLS protocol

[4] Despite recent high-profile attacks on the RC4 algorithm in TLS, its usage is still running at about 30% of all TLS traffic. We provide new attacks against RC4 in TLS that are focussed on recovering user passwords, still the pre-eminent means of user authentication on the Internet today. Our new attacks use a generally applicable Bayesian inference approach to transform a priori information about passwords in combination with gathered ciphertexts into a posteriori likelihoods for passwords. We report on extensive simulations of the attacks. We also report on a "proof of concept" implementation of the attacks for a specific application layer protocol, namely BasicAuth. Our work validates the truism that attacks only get better with time: we obtain good success rates in recovering user passwords with 2^{26} encryptions, whereas the previous generation of attacks required around 2^{34} encryptions to recover an HTTP session cookie.

2.4 Spoofing attack to the 3-D Secure protocol

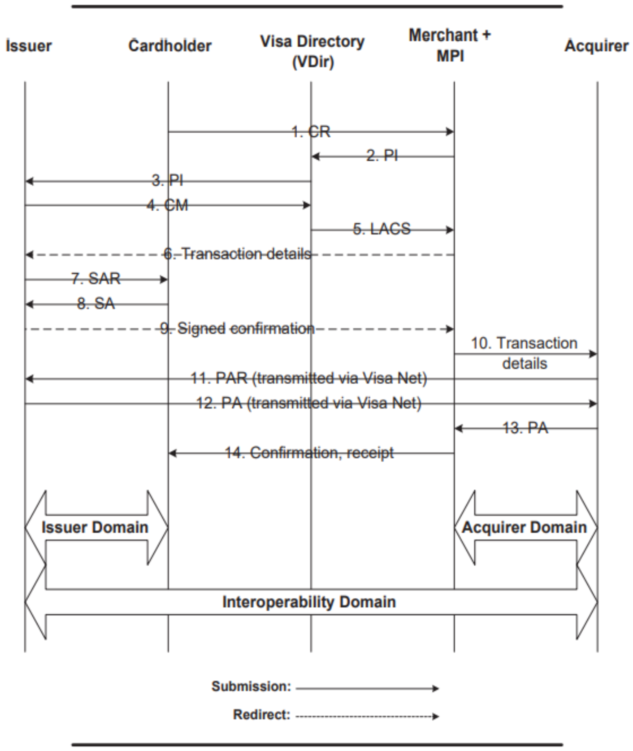

Fig. 1. The 3-D Secure transaction procedure

[5] This paper discusses the potential vulnerabilities associated with use of the 3-D Secure scheme. The authors show how 3-D Secure, and any other system based purely on the use of SSL and 'standard' web browsers, is vulnerable to 'spoofing' attacks: web spoofing and redirect spoofing.

2.5 Hill-climbing attack to the BioAPI protocol

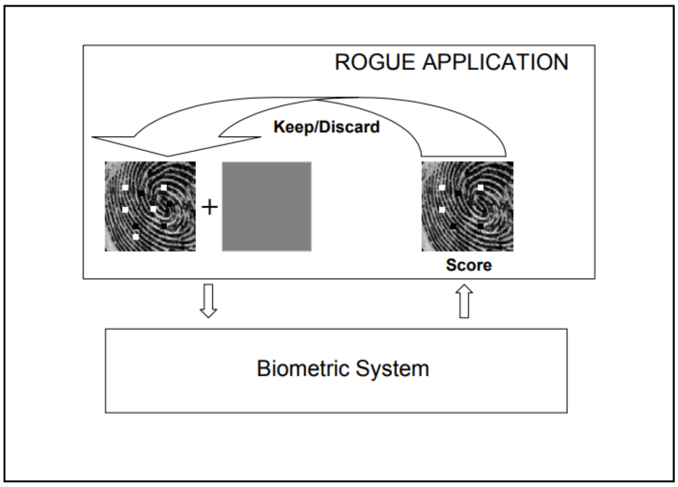

Figure 3. Steps in a hill-climbing attack

[6] Understanding the interface between biometric systems and general security systems is critical for the successful deployment of biometric technologies. The sensitivity of data passed between these two systems means that due care must be taken to avoid vulnerabilities such as identity, replay, and hill-climbing attacks. This is especially true as interfaces become standardized, such as with BioAPI, as these standard interfaces are available to developers and attackers alike. With this in mind, protection methods for data storage and transmission should be used to safeguard the systems.

In some applications, most particularly in the case of fused biometrics (i.e. face and fingerprint), the return of the score to the application provides useful information. In fact, this is one of the primary reasons that API's such as BioAPI support the return of the score in an operational biometric system. However, the developer of a biometric system needs to be aware of a potential vulnerability known as hill-climbing attack.

A hill-climbing attack can occur when an attacker has access to the biometric system and the user's template upon which he wishes to mount a masquerade attack. The attacker creates a simple rogue application that inputs the template along with a randomly generated image as input to the biometric system. The score returned by the biometric system is saved and the attacker randomly perturbs the image, retaining only those samples that positively increase the score. In this manner, the attacker can iteratively synthesis an image that produces a score that exceeds the threshold of the biometric system and use that image as input to the security system to which the original template belongs (see Figure 3).

3 References

[1] He, Weijia, et al. "Rethinking access control and authentication for the home internet of things (IoT)." 27th {USENIX} Security Symposium ({USENIX} Security 18). 2018.

[2] F. Callegati, W. Cerroni and M. Ramilli, "Man-in-the-Middle Attack to the HTTPS Protocol," in IEEE Security & Privacy, vol. 7, no. 1, pp. 78-81, Jan.-Feb. 2009, doi: 10.1109/MSP.2009.12.

[3] Mitrokotsa, Aikaterini, Melanie R. Rieback, and Andrew S. Tanenbaum. "Classification of RFID attacks." Gen 15693.14443 (2010): 14.

[4] Garman, Christina, Kenneth G. Paterson, and Thyla Van der Merwe. "Attacks Only Get Better: Password Recovery Attacks Against RC4 in {TLS}." 24th {USENIX} Security Symposium ({USENIX} Security 15). 2015.

[5] Jarupunphol, Pita. "A critical analysis of 3-D Secure." Proceedings of the 3rd Electronic Commerce Research and Development (E-COM-03) (2003): 87-94.

[6] Soutar, Colin. "Biometric system security." White Paper, Bioscrypt, http://www. bioscrypt. com (2002).