

Project Proposal Draft #1

Authentication and Secure Communication

Introduction

There is an increasing trend to delegate more responsibility to the mobile devices we surround us with. Due to the development of faster and cheaper devices we are no longer limited to temporal activities such as point-to-point communication. One can readily share data and as the phone platforms have grown more powerful also gather real-time information about their lives. The benefits are manifold but so are the risks.

As we use the mobile devices for more things we also increase the risk of exposing sensitive information such as location information. The risk increases yet again when we transfer the information to third-party application for later processing or data access from other devices. How we handle the challenge of protection user information without limiting the usefulness and ease-of-use of otherwise useful application is critical to future application developers.

It is very difficult to foresee and understand the needs of these mobile applications without real-world examples to be inspired from. We therefore base this study on the experience and feedback from an active team of developers and system designers' working on cell phone enabled applications. The developers perspective have since been complemented by the strict requirements for human studies (in specific the American standard for academical environments) and analysis of a wide variety of threats the users are exposed to.

To handle an increasing number of mobile users in a secure and efficient manor we present a tailored mobile user and application authentication framework. The framework allows identification and protection of individual users and will also allow third-party applications to connect and request permission to view or manipulate stored information.

User experience

The technical details of the systems are for most users irrelevant and users might not consider the often limited privacy protection guarantees the application gives. One might argue that as long as the owner of an application, e.g. a company or organization, is trusted users are willing to delegate significant responsibility to the application itself. There are two major things to learn from this. First, sensitive information will be shared as long as there is a positive benefit of doing so, even though the negative implications remain largely unknown. Second, a successful application can not burden the user with any unnecessary privacy protection details. In fact, the optimal scenario is an application that is able to perform the optimal configuration all by itself.

We therefore intend to provide a system that requires minimal action from the user yet supports various degrees of privacy protection for all systems. The weakest setting requires only partial encrypted communication whereas highly confidential information can benefit from the stronger setting, which requires end-to-end secure communication and guarantees encryption of all stored data.

The user authentication will be performed in the same manner for all platforms and will only require a

username and password combination. Hence, whether the user is uploading data from her cell phone or accessing statistics on a common website the login procedure will be the same.

To avoid having the user enter username and password every time data uploaded applications are launched on the phone a secondary, public-privacy key based protocol can be enabled. If enabled, the user login will be impersonated on subsequent login and allows secure transfer of data from the phone to the back-end system.

Architecture

The framework is build on top of the Kerberos and Public Key Infrastructure frameworks. Both are carefully designed to mitigate known attack types and are available in form of thoroughly tested libraries.

The system design relies in Kerberos for the authentication over untrusted connections. In practices this means authentication with third party applications and the preliminary authentication of new mobile devices.

Despite the fact the users are assumed to trust their applications the security system underneath does not. Therefore all user authentication with third party applications must be designed in such a way that the user does not have to expose credentials to others but the fully trusted security system. We accomplishes this by deploying a single-sign on web authentication interface. Hence, the user will be redirected to the security system own interface when entering username and password. An encrypted session token (containing the ticket) is then forward to the third party application, which proves the identity of the user by decrypting the session token.

Devices uses the Kerberos system upon the preliminary configuration of the device. In this state the user is still unknown to the system and is therefore unable to authenticate itself as part of the SSL negotiation. The client authenticate with the security system and if successful the a private key is pushed to the device. A one-way authenticated SSL connection is established for the key and certificate provisioning. This temporary tunneled is then teared down and replaced by a mutual authenticated tunnel in which the user now authenticates using the recently provisioned private key.

The security is further strengthened by limiting the private key for one-way communication only. Consequently the private key is only valid for transferring data to the server and is considered insufficient identification for accessing data. Even if the key was stole the potential damage would be minimal. A malicious user would at most be able to send obfuscated data and that only as long as the private key has not be added to the list of revoked certificates.

To support deferred upload, that is temporary buffering of data on the device, the system design supports a on-device data encryption. To accomplish this a dedicated public key for storage is pushed with the private key for communication. The device may then use the public key to encrypt any data before storing it to the flash memory. If the devices is compromised the data is unaccessible. To unlock the data the intruder has to get access to the private key, which is stored on the back-end system and may be protected by an additional symmetric pass-phrase. The 256 bit AES cipher is chosen for the purpose of encryption private-keys on the server. The size and run-tine of AES is neglible compared to the risk of exposing the private key in an attack.

The levels of protection is adjustable by the user. As the highest encryption level (256 bit AES) complies with the US governmental encryption standard for top secret information the system is designed to encompass the rigorous privacy requirements. While most will not have a use for the highest level of protection, some will.