tiCrypt Cryptographic Overview

Wednesday, February 3, 2016

Author: Thomas Samant, CMO
thomas@terainsights.com

tera insights

Public Key Encryption within tiCrypt

tiCrypt extensively uses public-key cryptography (RSA-2048) as its security foundation for secure and convenient communication between users, virtual machines, and the server.

Why Public Key Encryption?

Public key encryption is at the core of all security mechanisms that guard the internet:

- Allows checking the authenticity of a website through SSL certificates. This is how you know your are visiting your bank's website and not a phishing website.
- Allows establishing a secure communication channel with the web server. This is why you trust it when you provide sensitive information on the web.
- Allows digitally signing messages. This removes the need for physical signatures when creating contracts.

The use of public key encryption removes the need for passwords or secrets to authenticate, establish secure communication, or sign messages. The private part of the key is now the means to achieve security. The public part of the key is made widely available and stored in the clear on servers, client computers or even business cards. The magic of public key encryption is that everybody can encrypt using the public key but only the holder of the private key can decrypt. (Conversely, documents can be digitally signed using the private key, where the authenticity of the signature can be verified with the public key.)
It goes without saying that the larger the key, the more secure the data encrypted with it is. It is estimated that it would take half a million desktop computers running for 13 billion years (the age of our universe) to find the private key from an RSA-2048 public key.

A Key For Each User

Since public key encryption can do so many things, why not give one such key to each individual user. This would allow these users to securely:

- Authenticate/login: Occurs without a password. The server will use the public key to verify identity. A random challenge number is generated, encrypted with the public key and sent to the owner. To prove possession of the private key, the challenge has to be correctly decrypted.
- Share Files: Occurs without a file password/secret. Encrypting the file's symmetric key with the other user's public key ensures that only the recipient can decrypt the file.
- Encrypt Files and Drives: The keys for all encrypted files or drives the user owns are managed using the user's private key; only the owner of the private key can successfully decrypt the individual file and drive keys that allow further access.
- Issue certificates: These entitle other users to perform operations on the owner's behalf. Certificates are nothing else then a list of instructions digitally signed using the private key.

To enable all of these features, in tiCrypt each user gets an RSA-2048 key, generated when the user first registers – this is as strong as certificates used by banks, major websites and Fortune 500 companies.

With a Great Key Comes Great Responsibility

A key is only as strong as the mechanism to keep it secure. Much like your house or car key, should your private key get stolen, a copy can be made and all your data can be decrypted. Unlike your car key, the private RSA key is digital so it can be further guarded by a password. Take for example the tiCrypt key of Alin Dobra, our CEO, in the left panel:

- It is much longer than a password – not something you could remember. You need to carry it on a USB stick on the key ring next to your house key.
- It is protected by an AES-256 key derived from a password. To ensure that dictionary attacks are ineffective, a random salt, an initialization vector and 21444 rounds of AES-256 are used.
- Alin needs both the encrypted private key and the password to be able to access his files. Should any be missing, data cannot be accessed.
- Neither the password nor the encrypted private key are on the server. The server exclusively stores Alin’s public key.

Needless to say, the user needs to backup the encrypted key and remember the password protecting it. To avoid security breaches, by design, tiCrypt has no way to recover a lost key. It is user’s responsibility to backup and protect the key.