True Key™ by Intel Security

Security White Paper 2.0

Table of Contents

Introduction 1
Cryptosystem 1
Hardware based security 4
Communications security 5
Software architecture 5
Software engineering practices 6
Network and operational security 6
How to report a security issue 8
Conclusion 8

Introduction

True Key™ by Intel Security is the easier, safer way to unlock your digital world.

This White paper provides a technical overview of the comprehensive security measures the True Key app uses to protect your logins, wallet items, and safe notes.

Threat modeling and attack scenarios

The architecture of the True Key system is based on a comprehensive analysis identifying the potential threats or attacks that the system may face, focused especially on threats to the confidentiality, integrity, or availability of our users' passwords and wallet items. Attacks and threats of many types are included and analyzed, from malware attacks that attempt to capture a user's password data from the user's PC, to attempts by network hackers to break in to the True Key servers and data centers and steal the encrypted password database, to efforts by criminal organizations to infiltrate their members into an ISP or a Certificate Authority.

For each potential threat, specific measures have been identified to counteract or mitigate that threat. This analysis has guided the development of the comprehensive security measures throughout the True Key apps, as well as throughout the servers, systems, and processes with which the True Key apps operate. Many of these measures are described in this white paper.

Cryptosystem

Zero-knowledge design

The True Key app is based on a **zero-knowledge architecture** with respect to users' passwords and wallet items (including safe notes). This means that the True Key app's Internet-based servers have no access to these assets, and that no Intel employee has any access to these assets. This is achieved through the True Key app's underlying cryptosystem, the system of cryptographic measures that the True Key app uses to encrypt each user's assets and to authenticate users.

This section provides an overview of the True Key cryptosystem.

The Master Password

Every True Key user creates a Master Password (MP) during the registration process, which may be changed as frequently as the user wishes. To assist users in selecting a high-quality Master Password, the True Key app utilizes the **zxcvbn** password strength estimation library to provide immediate and realistic feedback to the user on the strength of their chosen password.

The Master Password is never stored or saved:

- It is never stored on the True Key servers
- It is never stored locally on any device
- It is never transmitted over the Internet or any local network

The Master Password is used as the basis for deriving one of the primary encryption keys used to encrypt the user's passwords and wallet assets, the Key Encryption Key (KEK). The KEK is generated from the user's Master Password with a random salt value, using a large number of rounds of a strong key derivation function, PBKDF2 with HMAC-SHA512.

User authentication

The user's Master Password is also used as the basis of the derivation of an Authentication Token, which is used as one of the factors required to authenticate the user on sign-in to the True Key app. This process works as follows:

- The Authentication Token is derived on the user's device using PBKDF2 with HMAC–SHA512, a strong key derivation function, from the user's Master Password together with a salt value that is guaranteed to be unique for each account (a different salt value from the one used to derive the KEK)
- The derived Authentication Token is sent over an encrypted TLS connection to the True Key servers
- The True Key servers hash the received Authentication Token with the bcrypt KDF (another strong key derivation function), and compare it to a stored bcrypt value. This stage of authentication is successful if the values match.

The True Key app performs additional authentication steps to further protect users' accounts. Several of these are described in the *Multi-Factor Authentication* section, below.

Client-side data encryption and decryption

When a user has successfully logged in to the True Key app as described above, the True Key servers synchronize the user's passwords and wallet data to the client in encrypted form. The True Key client must then decrypt the data that was sent. This is performed as follows:

- After authentication, the True Key app receives a copy of the user's data, encrypted with AES-256 in CCM mode, from the True Key servers.
- The True Key client computes the KEK from the user's unique 256-bit salt value (which it received along with the user's data) and the user's Master Password, using PBKDF2 with HMAC-SHA512
- After the Key Encryption Key has been computed, the True Key app can then unwrap the Content Encryption Key (a cryptographically random 256 bit key). This then allows the True Key app to decrypt and verify the integrity of the user's encrypted data using AES-256 in CCM mode.

When the user adds, changes, or deletes password or wallet data, the True Key app re-encrypts the data using the same process on the user's own device, and transmits the new encrypted data to the True Key servers, where it is protected for backup purposes and available for synchronization to the user's other devices.

The True Key servers store only strongly encrypted passwords and wallet items, encrypted for each user with a Master Password known only to the user plus a salt value that is unique for each user. For this reason, in the unlikely event that an attacker were able to gain access to the True Key servers, the attacker would face the laborious, resource-intensive task of a separate brute force attack attempt on each user's AES encrypted user data. In other words, an attacker could not attempt to brute force the data of all users simultaneously, but would need to invest the massive resources required for a brute force attack again for *each* True Key user.

Multi-Factor Authentication

The True Key app always authenticates the user with multiple factors. Trusted Device authentication (described below) is always used when logging in to the True Key Launchpad app, in combination with one or more additional authentication factors. This authentication method provides users with the most secure and convenient experience by providing a wide range of authentication options, including several biometric technologies (facial recognition, fingerprint recognition, and others to come), 2nd Device authentication using a mobile phone, a Master Password, and email based authentication. These options are also described in this section.

Trusted Device security

The True Key app provides additional security through the concept of a Trusted Device. A device is considered trusted by a user only if a user chooses to make it trusted. The process of marking a device as trusted results in the creation of cryptographic authentication material that can be used subsequently to positively identify the device.

The mechanisms used to securely generate and store this authentication material vary by device platform. On many Intel platforms these make use of strong hardware security-based security features – see the *Hardware based security* section below. On other platforms, the platform's native secure key stores, such as the iOS keychain, are used.

Trusted Device security is used in combination with biometric authentication. For example, any attempt to log in to the True Key app with facial recognition is rejected unless it is authenticated to be coming from one of the user's Trusted Devices.

When adding an additional device to their account, the user will first be required to authenticate on the new device. The True Key app will then also require the user to approve the request to add a new device using 2nd Device authentication, using one of the user's existing Trusted Devices. (In case a Trusted Device is not available, the user can choose to provide additional authentication using email.)

Biometric security

The True Key app provides support for biometric authentication on devices with the necessary sensors, including facial recognition, fingerprint recognition, and others to come.

Some biometrics may be used in place of frequent entry of the Master Password. These functions are protected by Trusted Device security (see above), and are implemented using either secure platform based security mechanisms (described in the *Hardware based security* section below) or using an additional encryption key and a key-splitting mechanism. Facial recognition authentication is performed either in a hybrid mode (performing facial recognition authentication both on the user's device and on the True Key servers), or in a server-based mode. The True Key servers will not release their portion of the required decryption key unless facial recognition authentication on the server is successful. Biometric templates for server-based facial recognition authentication (mathematical descriptions of biometric measurements, such as measurements of distances between facial features) are stored on the True Key servers in encrypted form, and are protected in storage by a Hardware Security Module (HSM), a device that provides strong protection for key material. Biometric templates cannot be used to reconstruct images of the user. No raw bio data such as facial images or iris images is stored.

The True Key app's biometrics, such as facial recognition, include anti-spoofing technologies, which use several techniques to protect against attempts to spoof a user's identity using photographs or videos. However, all biometric technologies are susceptible to spoofing to some degree. For this reason, the True Key app permits biometric authentication only from the user's own pre-selected Trusted Devices. If a biometric login attempt is received from a device which is not authenticated as one of the user's Trusted Devices, it is rejected.

The level of anti-spoofing protection desired by the user can be selected on a per-device basis in the True Key app's Lock Preferences settings.

2nd Device authentication

The True Key app provides the ability to perform additional authentication using a 2nd device, such as the user's mobile phone, that is one of the user's Trusted Devices. The True Key app sends a secure out-of-band notification to the selected 2nd device. The user receives this notification from the True Key app on their 2nd device, and swipes to verify the login. This means that only someone who has access to the designated mobile phone (in addition to knowing the user's Master Password and/or successfully completing biometric login) can log in to the True Key app.

Zero-knowledge account recovery

In case a user's Master Password is forgotten, the True Key app provides a secure mechanism for account recovery and password reset. Rather than relying on insecure "challenge questions", the True Key app uses a key-splitting mechanism which enables a user to recover their account and set a new Master Password if:

- The user has physical possession of one of their Trusted Devices, and
- The user can authenticate to the True Key servers

This works as follows:

- An Account Recovery Key is generated when the user's account is set up
- This key is split into fragments, which cannot be used separately
- One fragment is stored on each of the user's Trusted Devices, and another fragment is held by the True Key servers
- If a user needs to perform account recovery, the True Key servers authenticate the user using the True Key app's Multi-Factor Authentication capabilities, and if this is successful they release their fragment of the Account Recovery Key
- The released fragment can be combined with the fragment held on the user's Trusted Device, and the resulting combined Account Recovery Key can be used to reset the account

Hardware based security

The True Key app leverages, when available, hardware-based security capabilities of Intel® Architecture to enhance the level of protection and to provide higher levels of security for the user's passwords and wallet items. Some of these capabilities are described in this section.

Intel Identity Protection Technology (IPT)

Intel® Identity Protection Technology provides a hardware root of trust, giving proof of the identity of a unique PC, and is used by the True Key apps to support Trusted Device authentication.

Intel Identity Protection Technology is a feature found on Intel processors since 2011, and is available today on over 500 million PCs and other devices with Intel® Core™, Intel® Xeon®, Intel® Pentium®, Intel® Celeron®, and Intel® Atom™ processors.

IPT employs a dedicated and protected processing environment that is separate from the main processor. The authenticity of this environment is attested (verified) using the Enhanced Privacy ID (EPID) feature of the processor. The environment also contains a real-time clock that is separate from the one used by the host system.

When the user sets up their True Key profile, IPT is initialized using the SIGMA (Sign-and-MAC) protocol, which performs an authenticated Diffie-Hellman key exchange to establish a shared secret between the user's device and the True Key servers.

Using this shared secret and the dedicated time clock, IPT is able to generate time-based one-time password (OTP) tokens. Every subsequent communication between the True Key app and the True Key servers must contain a valid OTP token or it will be rejected.

This provides a strong hardware basis for the True Key app's Trusted Device authentication.

Intel Software Guard Extensions (SGX)

Intel® Software Guard Extensions (Intel® SGX) are a new set of instructions added to the Intel Architecture instruction set that provide a Trusted Execution Environment (or TEE) that increases the security of application software. SGX works by providing a CPU-based mechanism that executes critical software inside an enclave to protect from attack by malware and by higher-privilege software, in the event that system software has been compromised.

SGX is currently available in 6th generation Core branded processors.

The True Key app leverages SGX by using an enclave to protect the sensitive processing pertaining to password and wallet item encryption and decryption, protecting it from attacks, even if other parts of the PC have been compromised.

Communications security

Pervasive TLS

All communications between the True Key app and the True Key servers are encrypted using TLS, with carefully chosen cipher suites, as an additional layer of security. The TLS protocol operates as follows:

- The client and the server negotiate to choose the best cipher and hash algorithm available
- The server transmits its digital certificate
- The client verifies that the server's certificate is signed by a trusted Certificate Authority (CA), and may contact the CA to verify the certificate
- The client and the server negotiate an ephemeral session key using the Elliptic Curve Diffie–Hellman key agreement protocol (ECDHE)
- Alternatively, if the client doesn't support Diffie-Hellman key agreement, the client generates and encrypts a random number with its public key and transmits the encrypted number to the server, and both sides use this number to generate the session key
- The new session key is used to encrypt all subsequent traffic between the client and the server

The True Key servers properly handle the negotiation of TLS cipher suites, including mitigations for the BEAST, CRIME, BREACH, POODLE, Logjam, and related attacks.

HTTP Strict Transport Security (HSTS)

The True Key app uses HTTP Strict Transport Security (HSTS) in order to protect against TLS stripping (HTTP downgrade) attacks in the event that a user connects to the True Key web site from an untrusted network. Additionally, the True Key app communicates with its servers exclusively using TLS.

JSON Web Token

JSON Web Token (JWT) is a compact means of securely exchanging authenticated data between two parties. The claims in a JWT are encoded as a JSON object that is digitally signed using JSON Web Signature (JWS).

The True Key app makes use of JWT instead of relying on HTTP cookies to persist session state. This has several benefits including not being vulnerable to Cross-Site Request Forgery (CSRF) attacks, because the JWT is not returned automatically by the browser in every request.

Additionally, the JWT for the True Key Launchpad session is protected by being held an execution context that is completely isolated from the main JavaScript execution of the Web browser, thus protecting against Cross-Site Scripting (XSS) attacks.

Software architecture

Least-privilege architecture

Each component of the True Key app is designed to operate with the lowest level of system privilege required in order to perform its necessary functions.

Process isolation and protected execution contexts

The True Key apps make extensive use of the process isolation models available on each platform in order to isolate security related processing from external and potentially hostile code.

Selection of cryptographic primitives

The True Key app makes use of only the strongest and most trusted cryptographic primitives. These include:

- AES-256 in CCM mode
- RSA with 2048-bit keypairs
- PBKDF2 with HMAC-SHA512
- bcrypt KDF

Selection and validation of modules

Where the True Key app makes use of third party software modules and code libraries, only the highest-quality components, which have been subjected to extensive third-party and internal review, are used.

Platform-specific security mechanisms

The True Key app makes use of the best security mechanisms available on each platform. These include:

- On Windows, the True Key app uses the Windows Biometric Framework (WBF) to support authentication devices such as fingerprint readers
- On Windows, the True Key app makes use of the

additional cryptographic protection provided by the Data Protection API (DPAPI), which enables encrypted material to be bound to a Windows user account

- On both iOS and MacOS, the True Key app uses the secure keychain mechanism provided by the platform for secure storage of sensitive data
- On iOS, the True Key app integrates with TouchID to perform secure biometric fingerprint authentication
- On Android version 6 and later, the True Key app uses the secure keystore provided by the platform for secure storage of sensitive data

Software engineering practices

Software Development Life Cycle (SDLC)

The True Key software is developed within a Software Development Life Cycle (SDLC) framework defined by Intel and Intel Security that:

- Specifies requirements for security training for all members of the development team
- Defines mandatory security processes at each stage of software design, development, review, testing, deployment, and maintenance
- Establishes standards for security-sensitive software architecture and implementation, and
- Provides for mandatory oversight of these processes

Static Application Security Testing (SAST)

All the True Key software components are subject to a regular cycle of Static Application Security Testing (SAST), also known as static analysis, in order to provide early identification of any implementation-level weaknesses. The SAST process is supervised by skilled security experts who are independent of the software engineering team.

Dynamic Application Security Testing (DAST)

All the True Key software components are subject to a regular cycle of Dynamic Application Security Testing (DAST), supervised by skilled security experts, in order to provide additional indications of any software weaknesses.

Code review

Each of the True Key software components is reviewed in depth by software security experts who are independent of the software engineering team that developed the respective component. No component becomes part of the True Key product without approval from the product security team.

Network and operational security

Corporate security policies and standards

All aspects of the True Key app's operations are governed by the comprehensive security policy and standards frameworks of Intel and Intel Security. These standards clearly define roles and responsibilities, management's engagement towards security, the security requirements with which every employee must comply, and technical standards for secure software development, server and network hardening, etc.

Data centers

All of the True Key servers are located in highly secure datacenter facilities with 24/7 security guard presence and biometric security for entry.

Certifications

The True Key data centers have received the following Certifications and Third-Party Attestations:

- Validated as a Level 1 Service Provider under PCI-DSS
- Certified against the Common Security Framework (CSF) from the Health Information Trust Alliance (HITRUST) and has been certified for HIPAA compliance
- SSAE 16 SOC 1 Type 2, SOC 2 Type 2, SOC 3 and ISAE 3402 reports, demonstrating the viability of the security control program over time
- Received a certificate of approval for our control program against the ISO/ IEC 27001:2005 standard for Information Security Management Systems

Server hardening

All of the True Key servers are hardened to minimized configurations that run only the required components and services, in order to minimize potential attack surface.

Network isolation

The True Key server environments are segregated into multiple isolated network tiers, according to the nature of the processing performed in each tier. Sensitive systems are strictly demarcated and protected by additional layers of security. Each network tier is isolated from others and only the minimum required traffic flows are permitted between tiers.

Logical security systems

The following additional logical security services provide protection for the True Key server environments:

- IP Reputation Management systems are in place to quickly compare source IP addresses to known and dangerous reputation lists in order to instantly deny access to known attackers
- Log Monitoring and Management systems are in place to detect and prevent unauthorized access to the True Key systems. All logs are safeguarded to protect their integrity and allow correlation of events for enhanced monitoring
- Intrusion Detection systems (IDS) are deployed throughout the environment and are regularly monitored
- Ongoing Vulnerability Management scanning is performed to ensure that software vulnerabilities and configuration errors do not affect the True Key server environments
- Web Application Firewalls (WAFs) are used to inspect all web traffic detect and block attacks such as XSS (Cross-site Scripting) and SQL Injection
- Next-Gen Firewalls with egress and ingress filtering are in place to identify all traffic flows and to permit only authorized network traffic to enter or leave the True Key server environments
- DDoS/DoS protection systems – Layer 3, 4 and 7 DoS protection is in place to safeguard resources and bandwidth for legitimate customer traffic

Secure remote management

Management of the True Key servers is performed exclusively through strongly encrypted mechanisms with multiple strong authentication factors in order to enhance security and ensure accountability for all administrative activities. Role-Based Access Control (RBAC) is enforced on all systems. No access is granted to any employee except when required according to operational need and at the least level of privilege necessary to perform the duty.

Patch management

All servers and applications are kept up to date with the latest tested patches in the production environment. Prior to deployment, all patches are tested in pre-production and development environments to ensure continuous availability of the production environment.

Change management

All of the True Key servers are subject to strict change management programs in order to minimize the risk of corruption of the production environment. The process ensures that all changes are tested and approved prior to being deployed in production.

Third party testing and validation

In addition to ongoing security testing by highly-skilled teams within Intel and Intel Security, the True Key systems are regularly validated by third-party experts.

How to report a security issue

If you believe you have identified a security issue with the True Key app, please contact the Intel Security Product Security Incident Response Team (PSIRT) by email at *PSIRT@IntelSecurity.com*. A PGP key is available to enable you to communicate securely with the PSIRT team.

Conclusion

The True Key app performs many complex security functions, as described throughout this white paper, but our ultimate goal has been simple: to provide you with the easier, safer way to unlock your digital world. We hope this white paper has been helpful in showing you how we accomplish this goal.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at http://ipt.intel.com/detect.