

Securing and Managing Trust in Modern Computing Applications

Andy Sayler

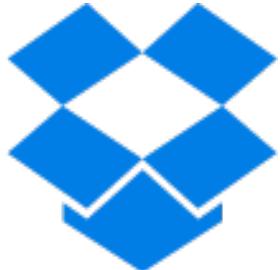
Dissertation Proposal
04/09/15

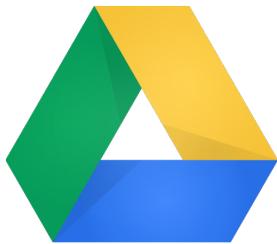
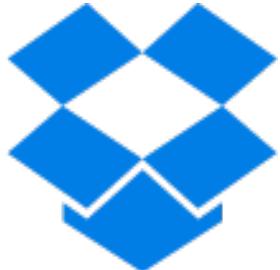


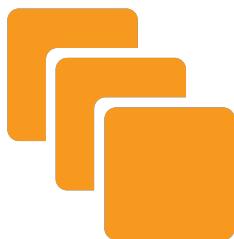
University of Colorado **Boulder**

Introduction

Where do we
store and process
our **data** today?







Who must we trust?

Google

Google

amazon

Google

amazon



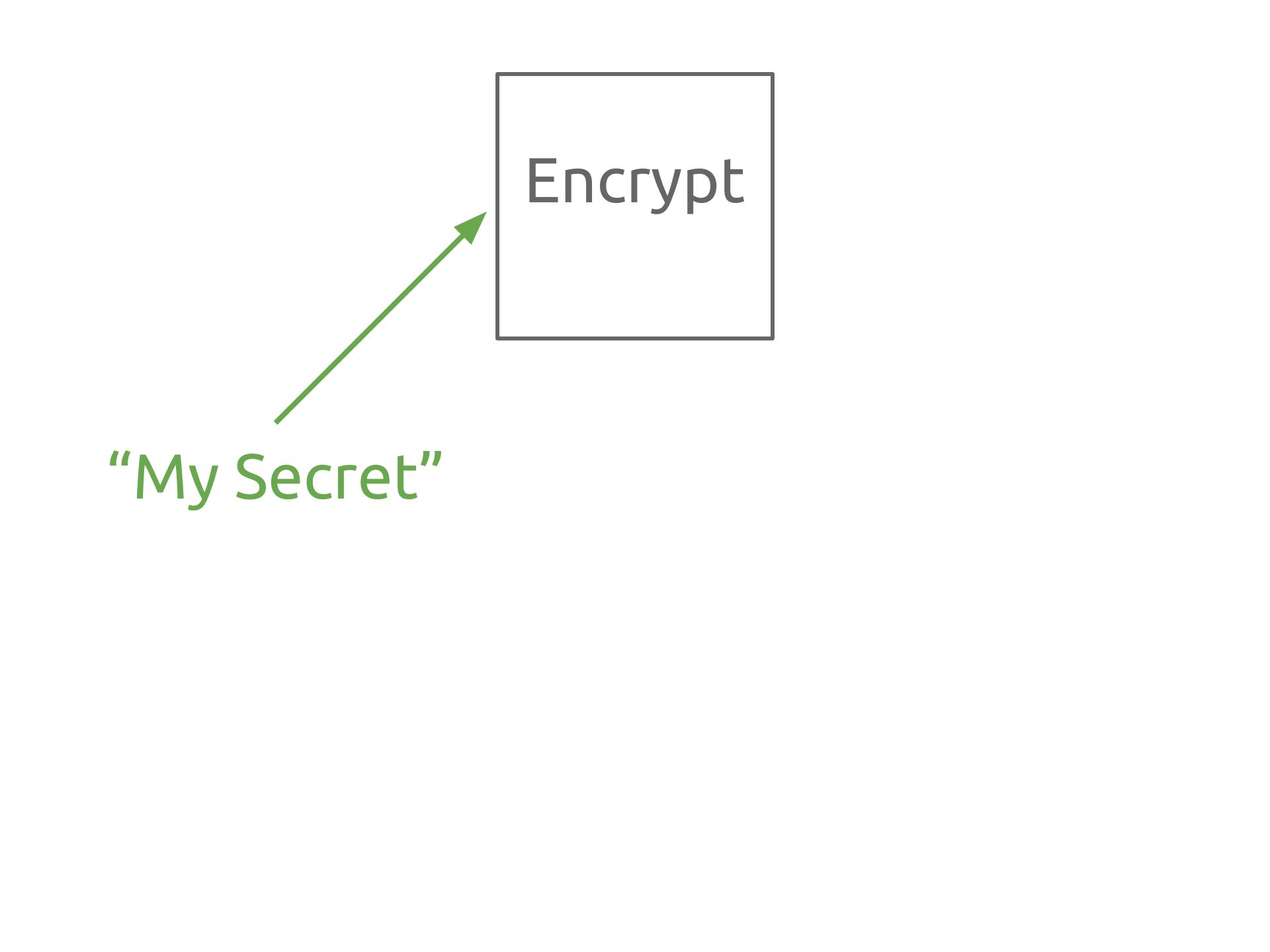
How can we **secure**
and **control** our data?

How can we **secure**
and **control** our data?

*(even in the presence **third parties**)*

Client-Side Encryption?

“My Secret”



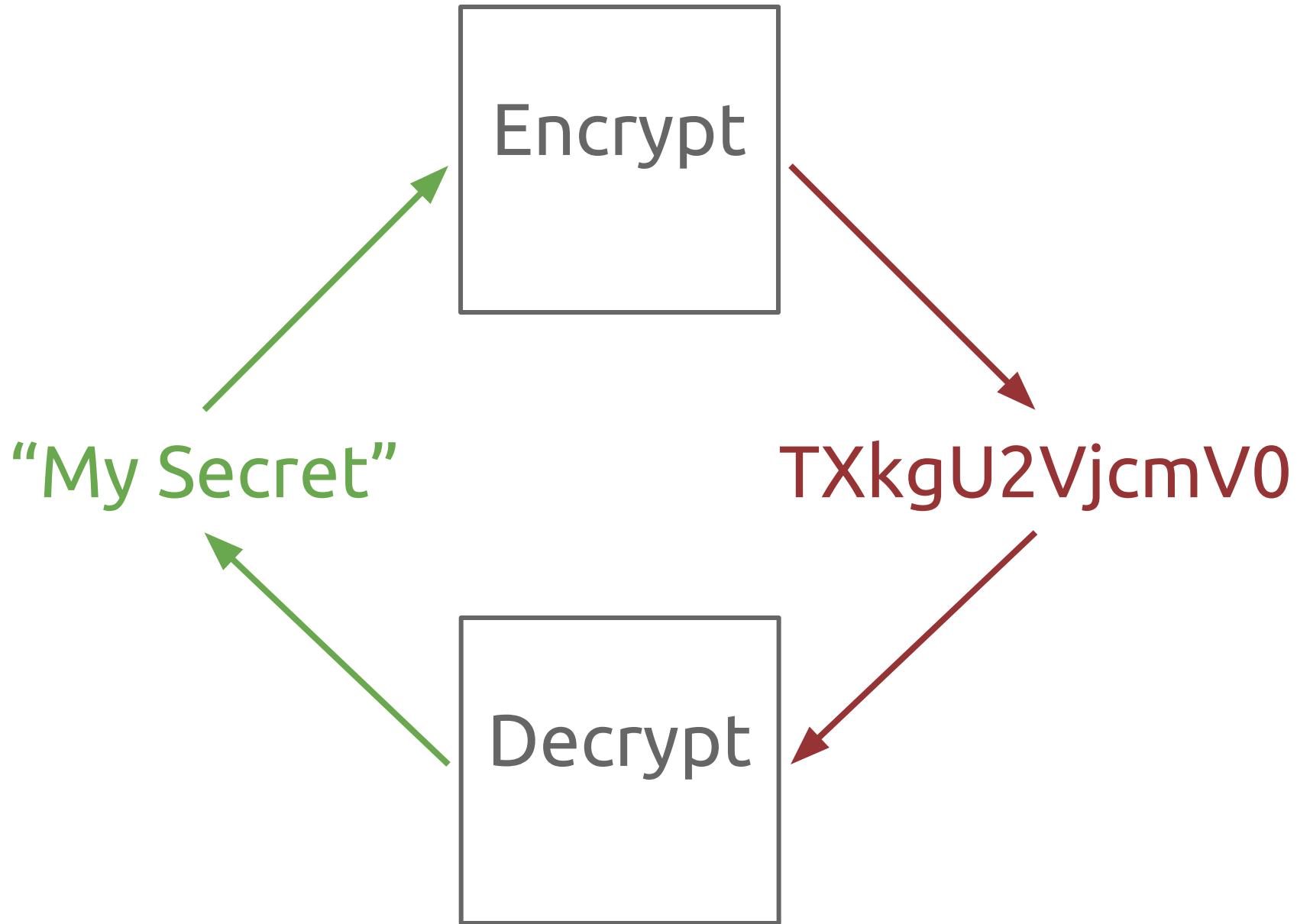
Encrypt

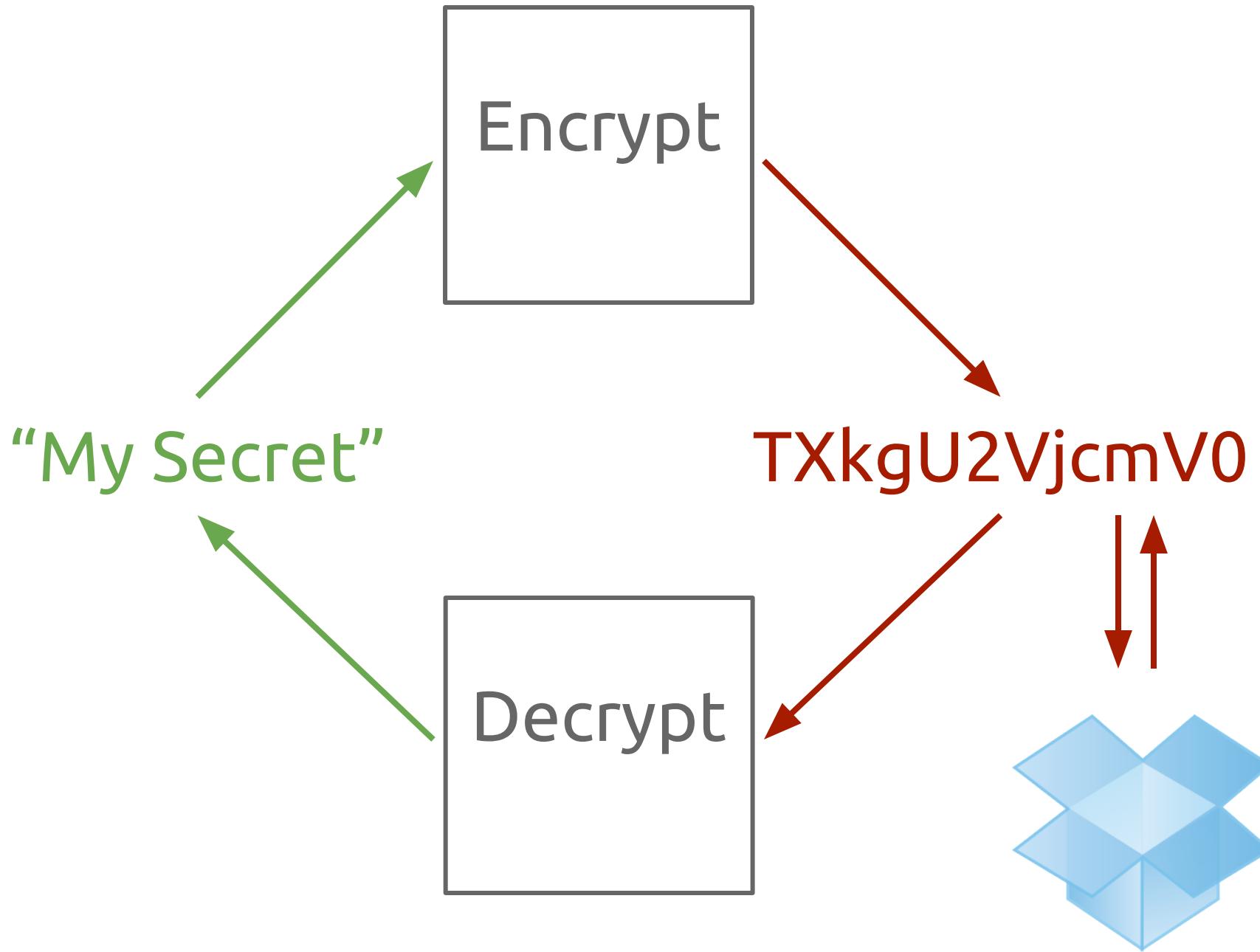
“My Secret”

“My Secret”

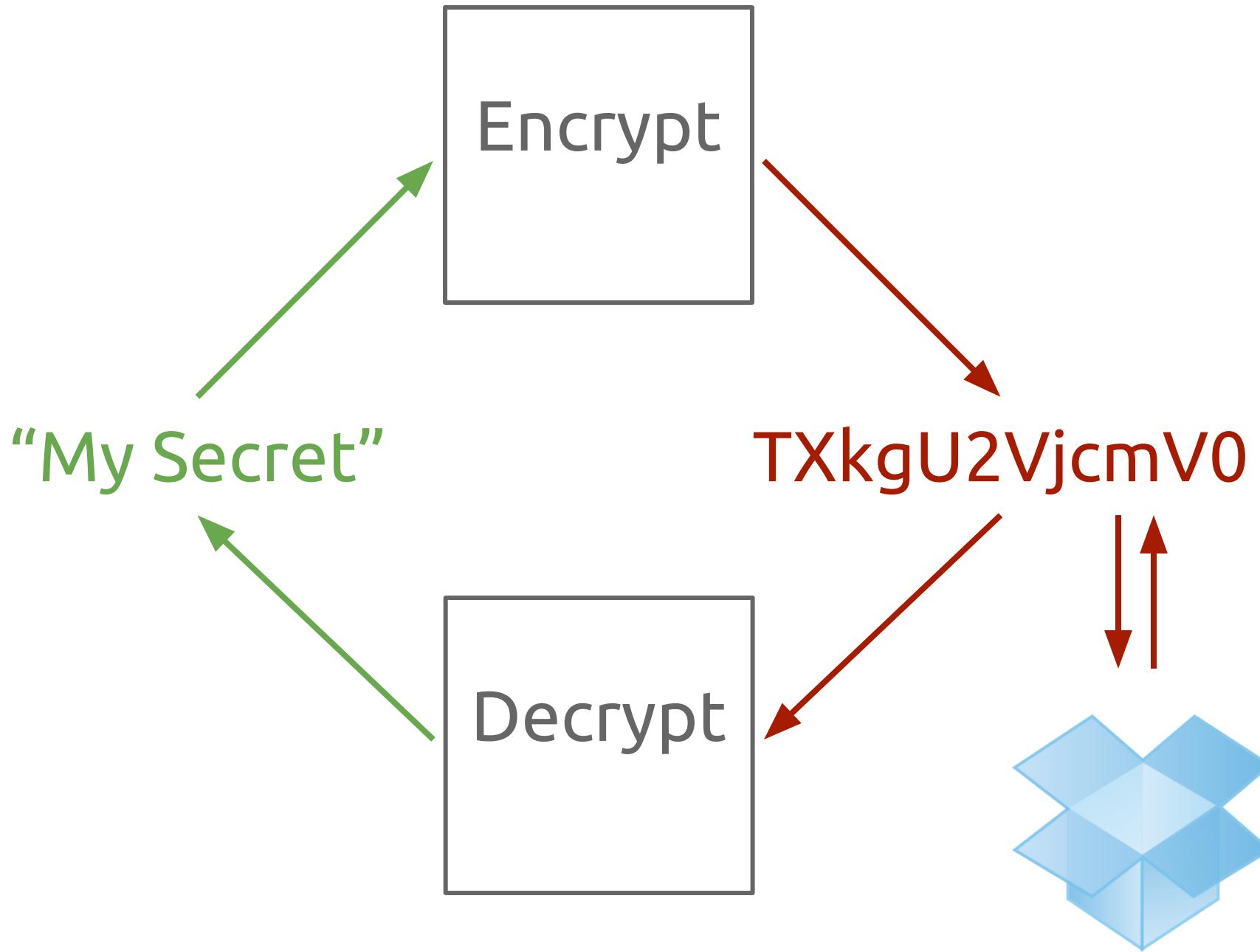


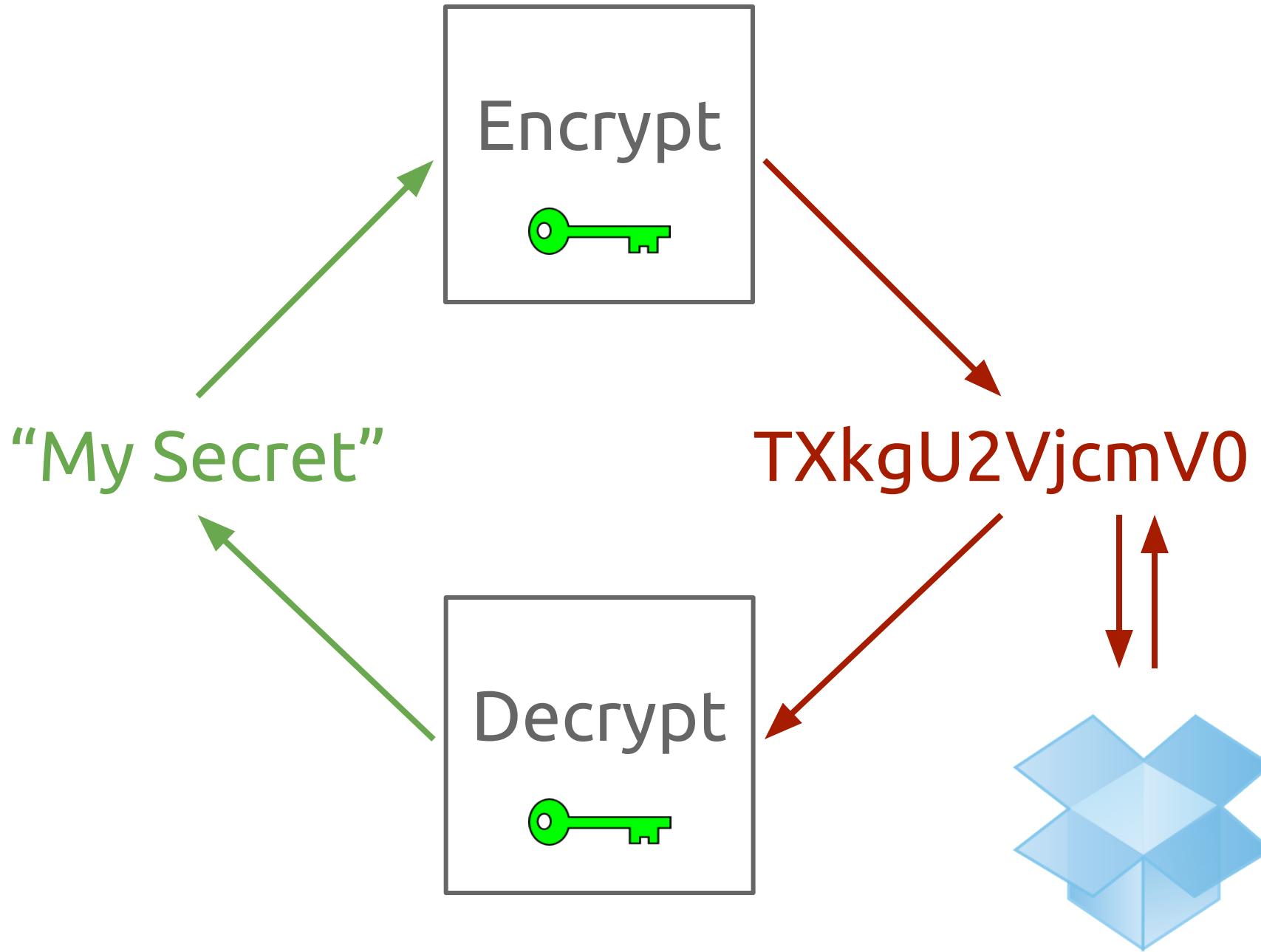
TXkgU2VjcmV0

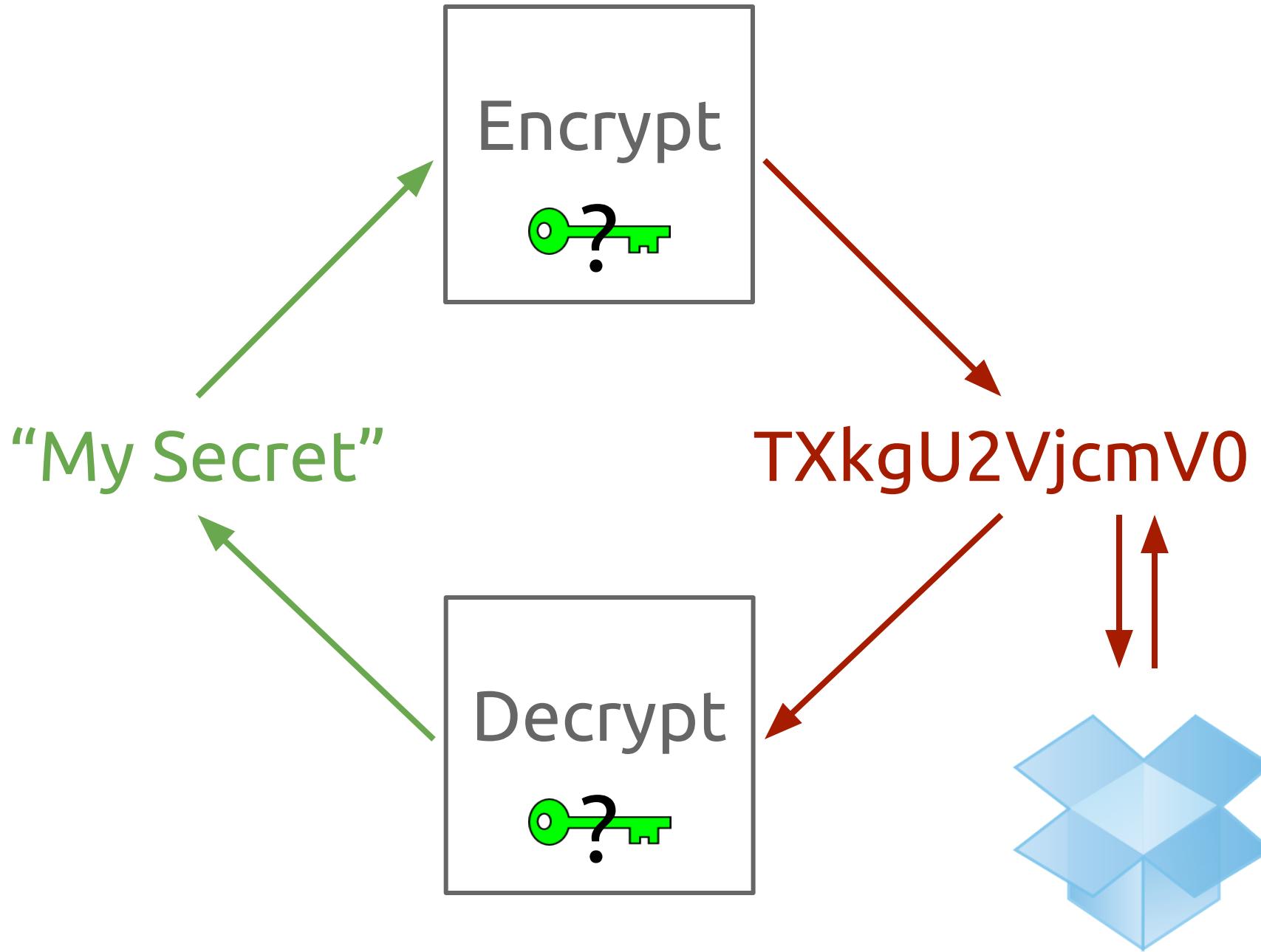




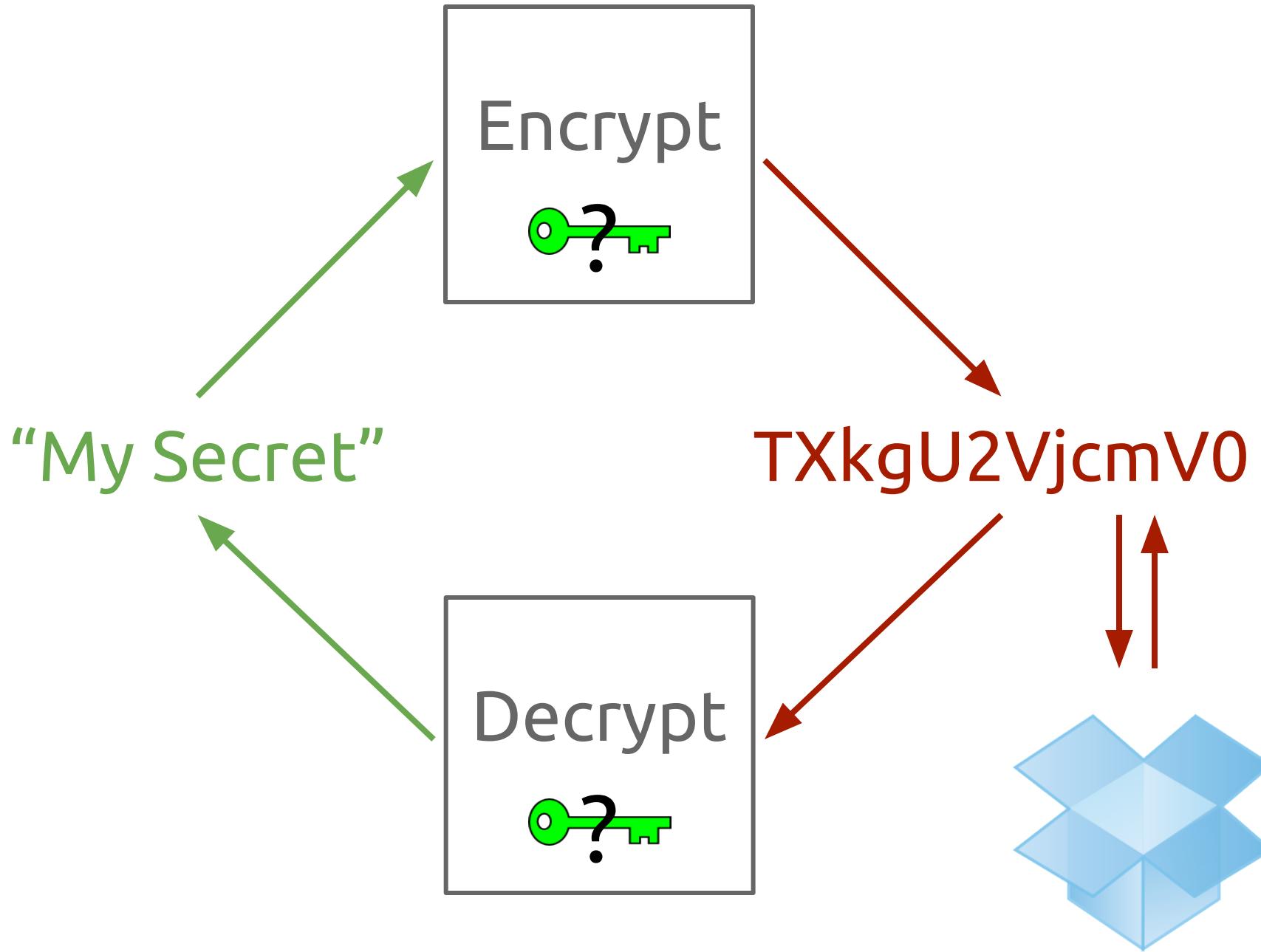
But what about the **keys**?

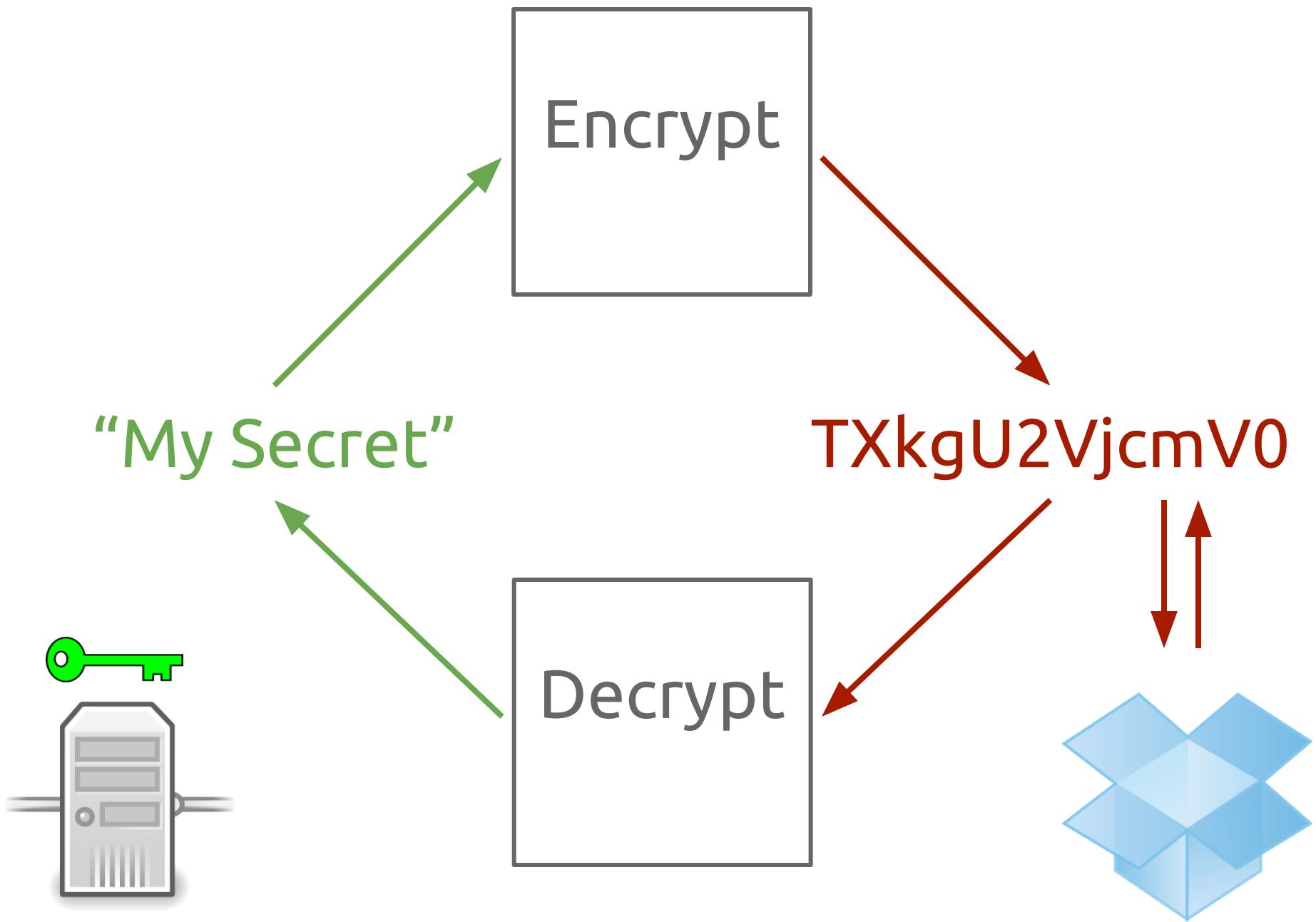


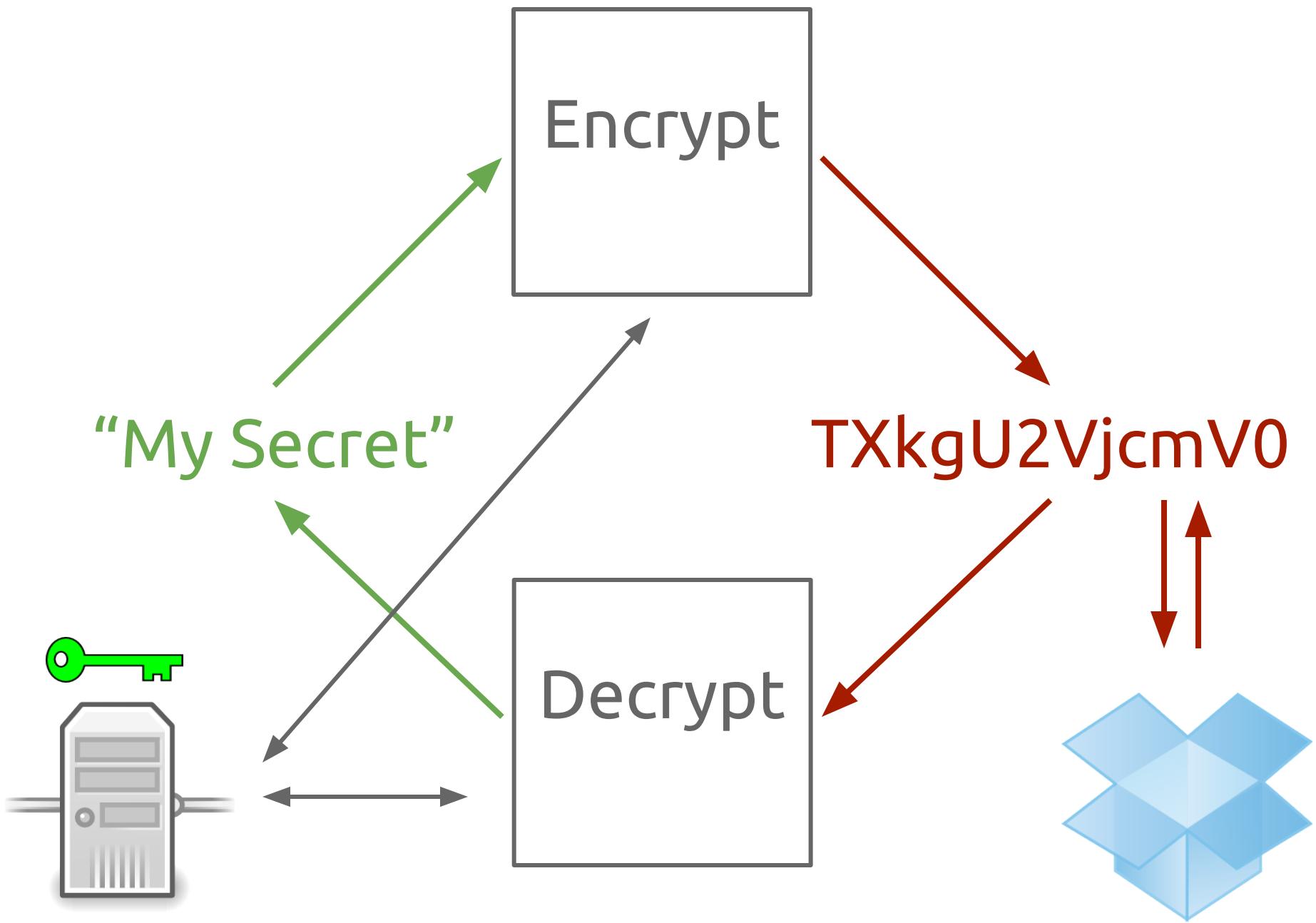




Preview: Secret Storage as a Service







Goals

- + Quantify and analyze third-party trust exposure inherent in modern applications

- + Quantify and analyze third-party trust exposure inherent in modern applications
- + Provide primitives for minimizing, managing, and monitoring third party trust exposure

- + Quantify and analyze third-party trust exposure inherent in modern applications
- + Provide primitives for minimizing, managing, and monitoring third party trust exposure
- + Use primitives to create security and privacy enhancing systems for modern applications

Background

Cryptography

Cryptography

Symmetric Asymmetric

Encryption Authentication

Verification Secret Sharing

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Usability

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Usability

PGP

Key Management

Misconfiguration

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Usability

PGP

Key Management

Misconfiguration

Access Control

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Usability

PGP

Key Management

Misconfiguration

Access Control

Unix Perms

ACLs

Role-Based
(RBAC)

MAC vs DAC

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Storage

Usability

PGP

Key Management

Misconfiguration

Access Control

Unix Perms

ACLs

Role-Based
(RBAC)

MAC vs DAC

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Storage

Distributed
File Systems

Cryptographic
File Systems

Failure Resistance

Internet-Scale

Usability

PGP

Key Management

Misconfiguration

Access Control

Unix Perms

ACLs

Role-Based
(RBAC)

MAC vs DAC

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Storage

Distributed
File Systems

Cryptographic
File Systems

Failure Resistance

Internet-Scale

Usability

PGP

Key Management

Misconfiguration

Access Control

Unix Perms

ACLs

Role-Based
(RBAC)

MAC vs DAC

Cloud Computing

Cryptography

Symmetric

Asymmetric

Encryption

Authentication

Verification

Secret Sharing

Storage

Distributed
File Systems

Cryptographic
File Systems

Failure Resistance

Internet-Scale

Efficiency

OPEX vs CAPX

IaaS

SaaS

PaaS

Usability

PGP

Key Management

Misconfiguration

Access Control

Unix Perms

ACLs

Role-Based
(RBAC)

MAC vs DAC

Cloud Computing

Virtualization

Commoditization

Challenges

Modern Demands

Modern Demands



Third-Party Solutions

Modern Demands



Third-Party Solutions



Security and Privacy Concerns

Modern Demands



Third-Party Solutions



Security and Privacy Concerns



New Solutions?

Use Cases

Multi-Device File Access



Multi-Device File Access



Multi-Device File Access



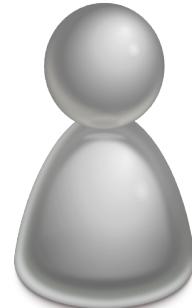
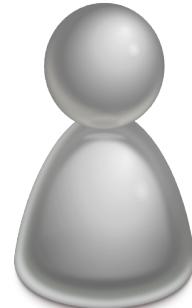
Cloud File Storage



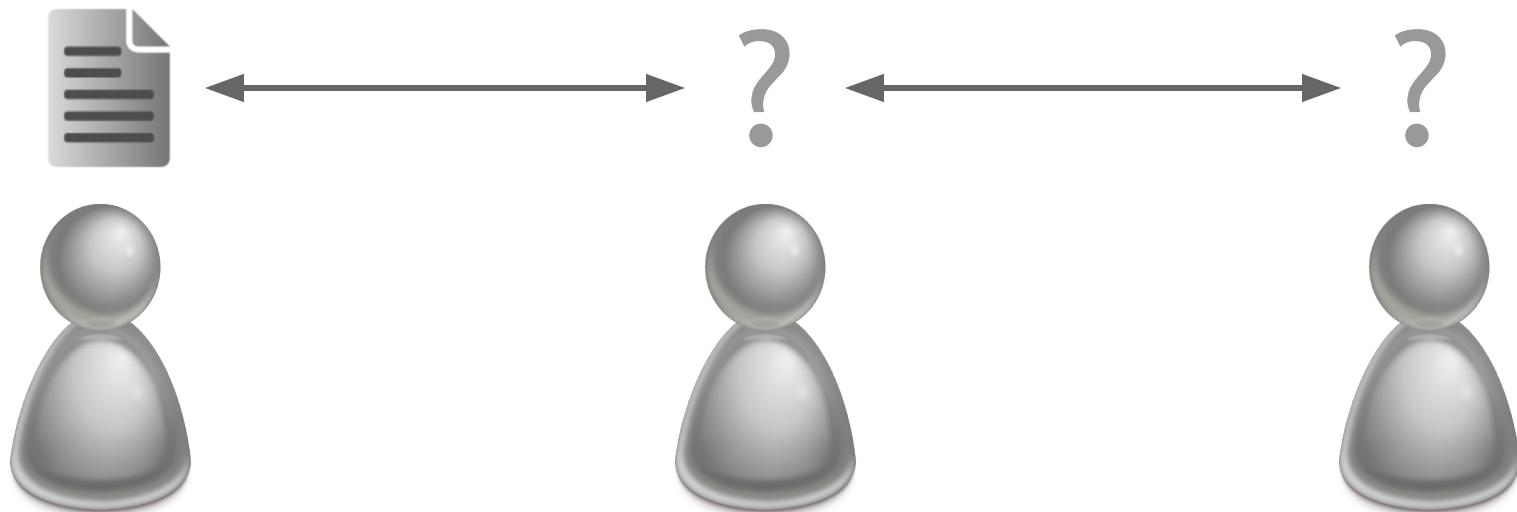
Multi-Device File Access



Multi-User File Sharing



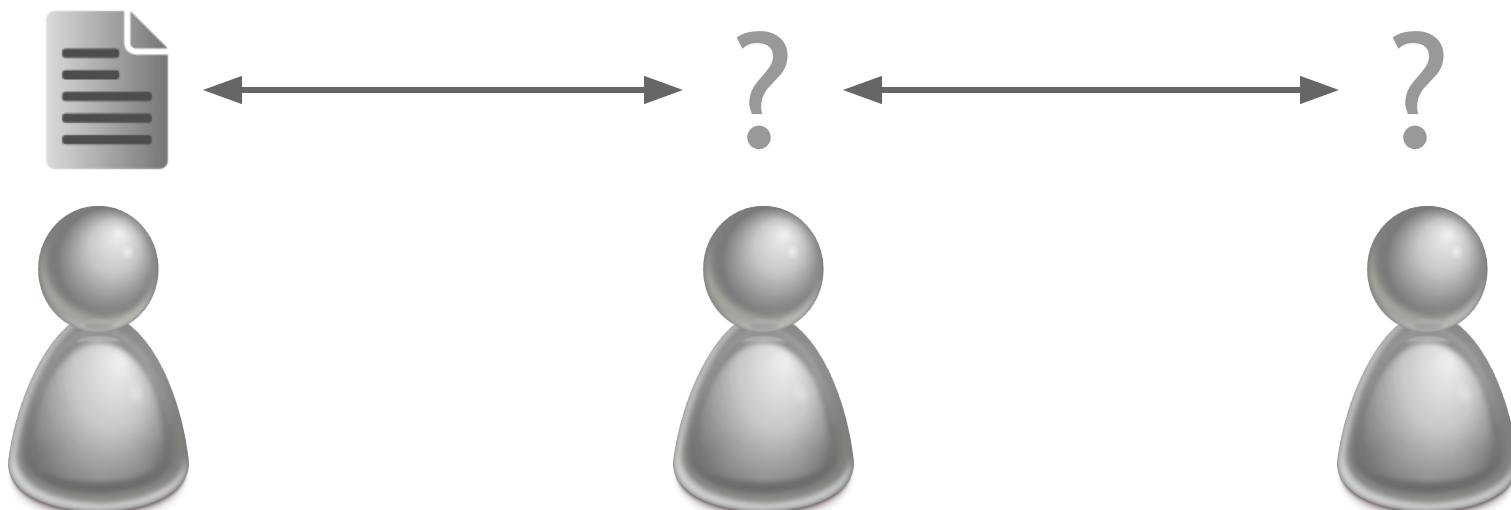
Multi-User File Sharing



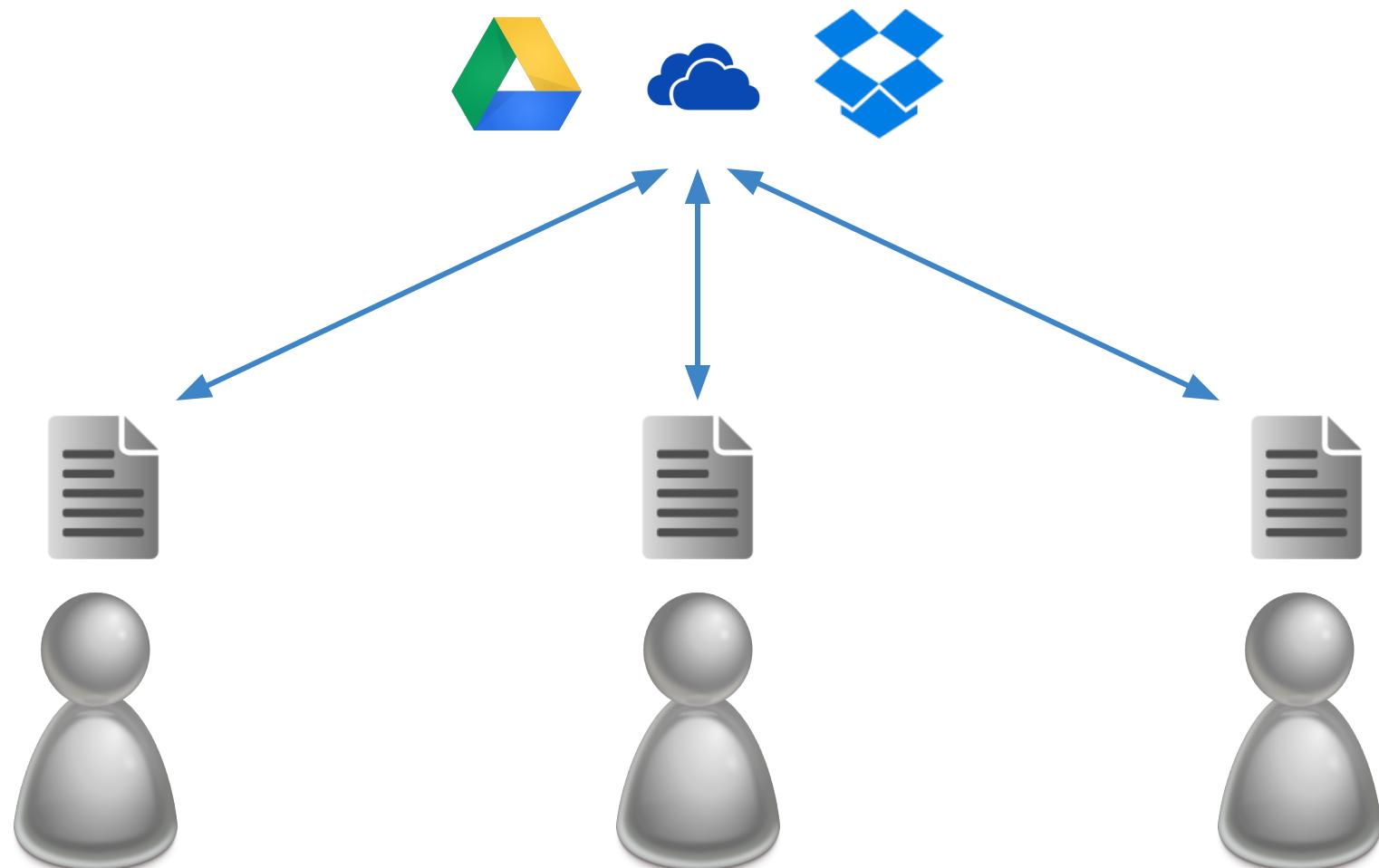
Multi-User File Sharing



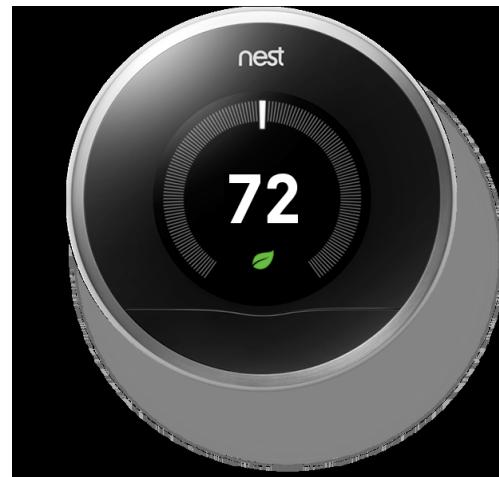
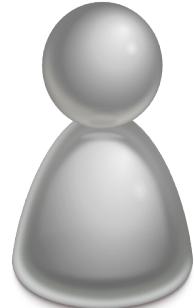
Cloud File Storage



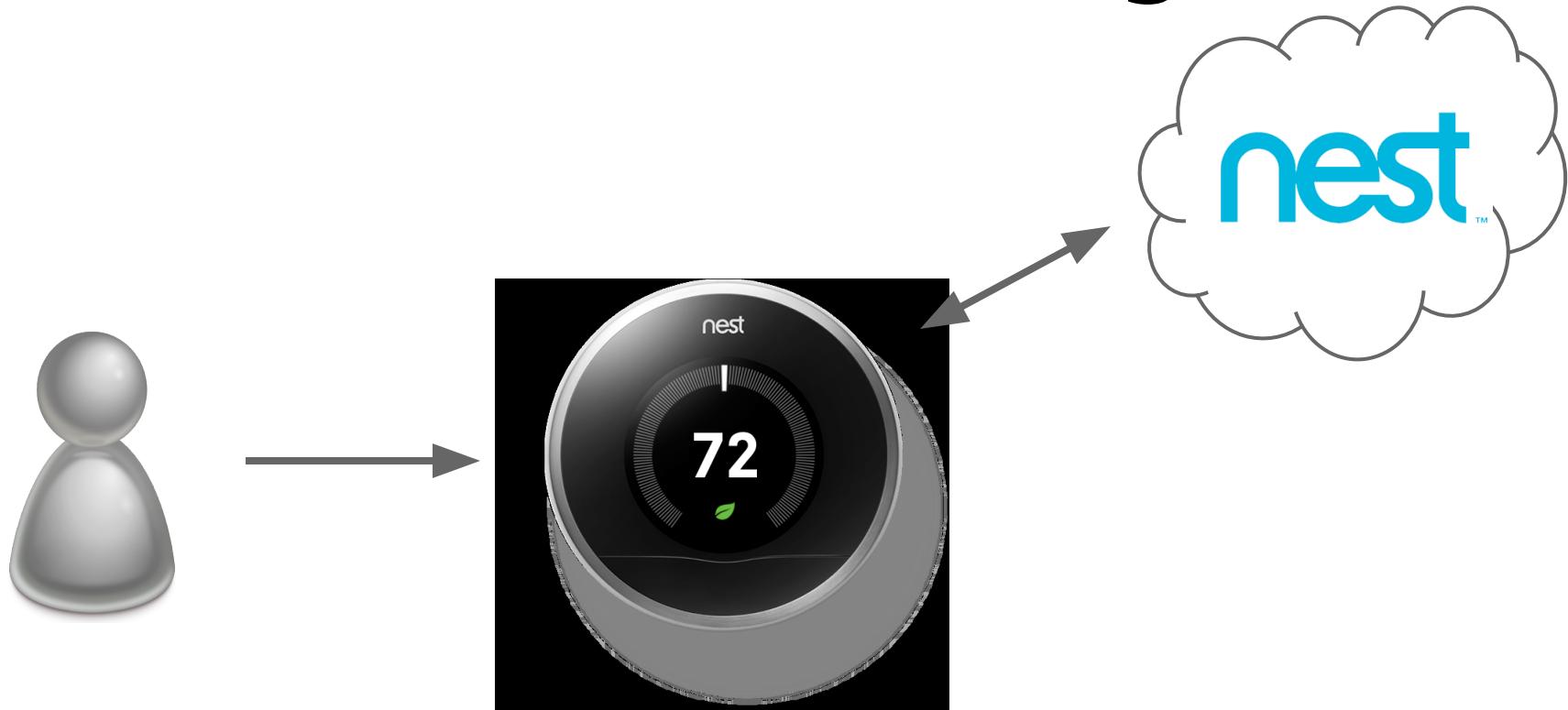
Multi-User File Sharing



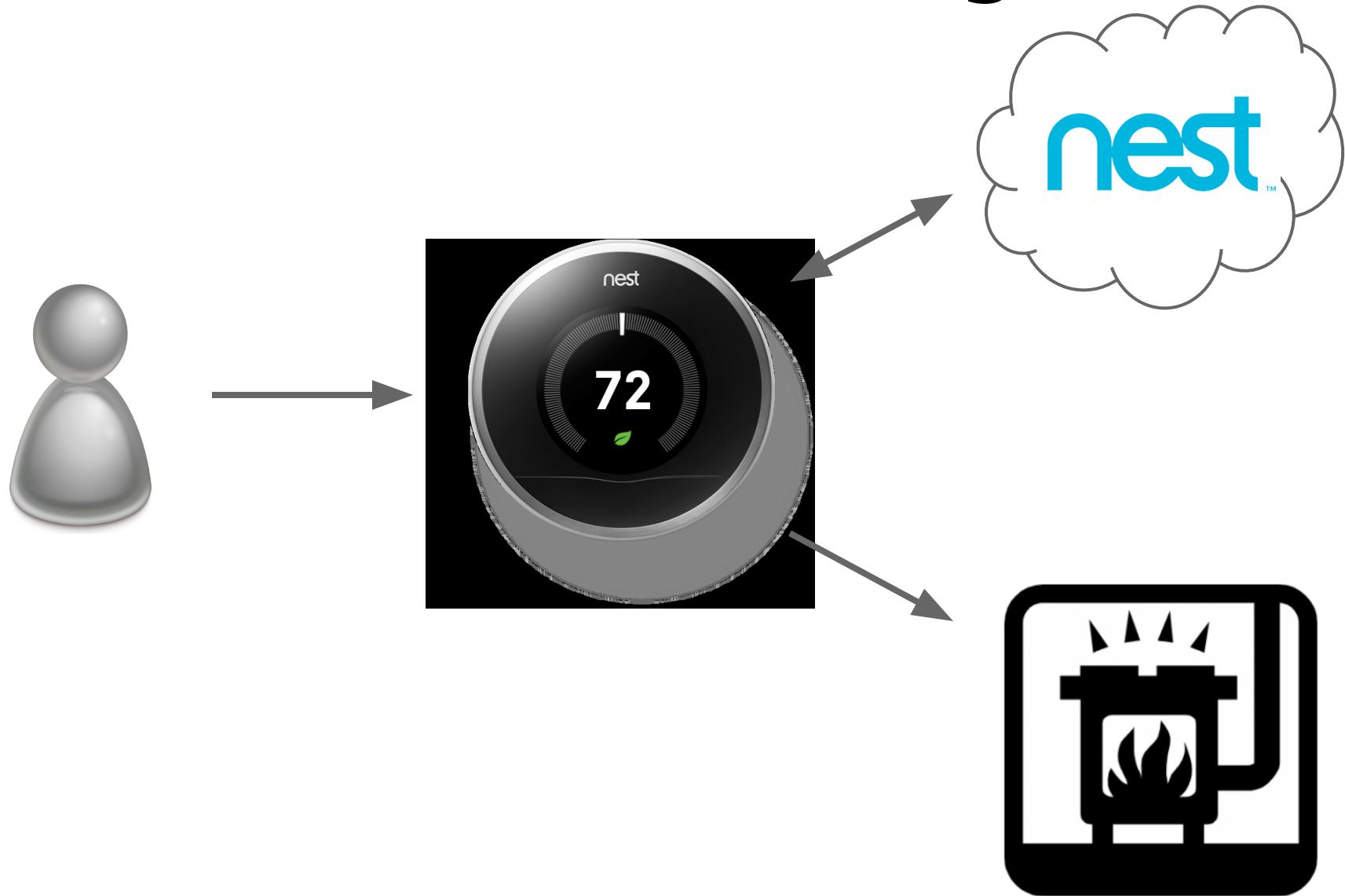
Data Processing



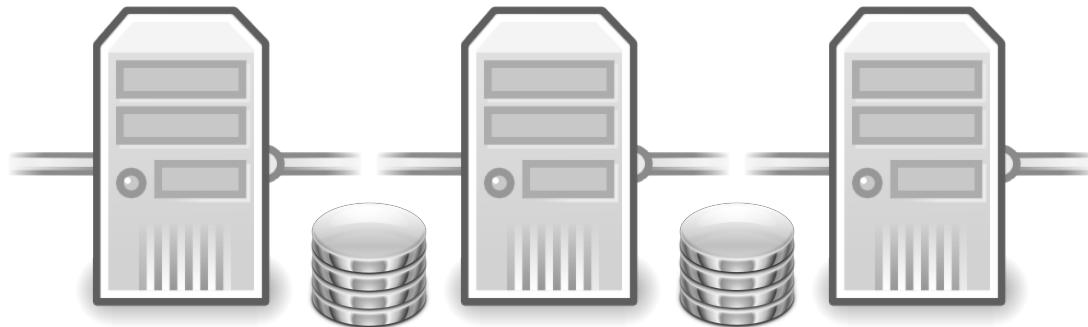
Data Processing



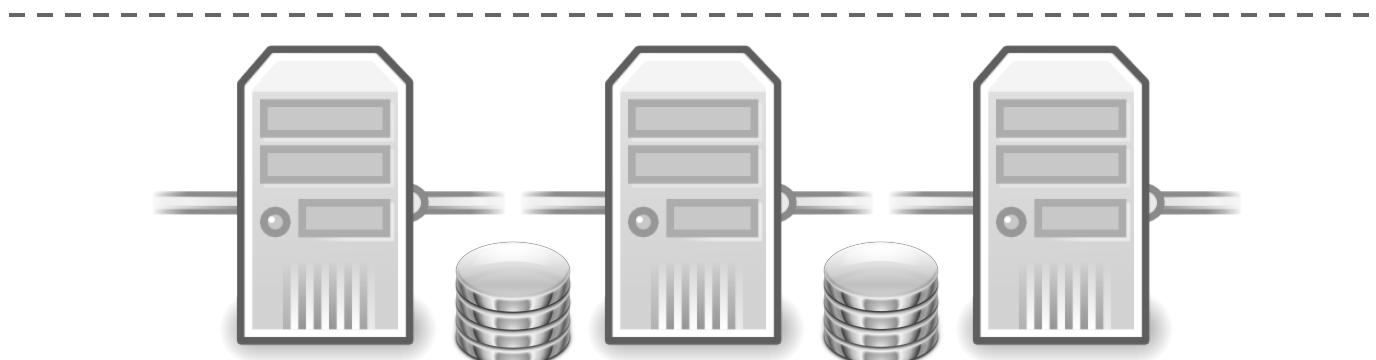
Data Processing



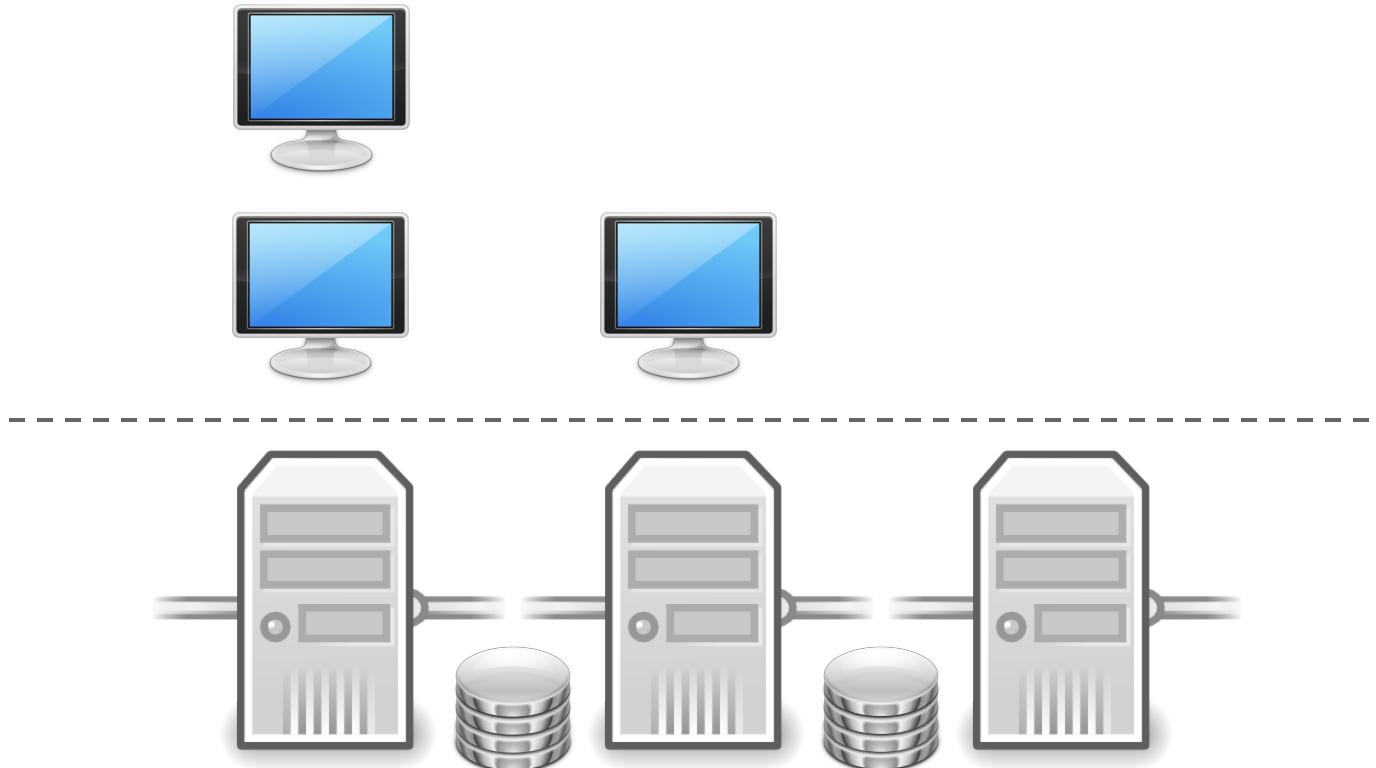
Cloud Infrastructure



Cloud Infrastructure



Cloud Infrastructure



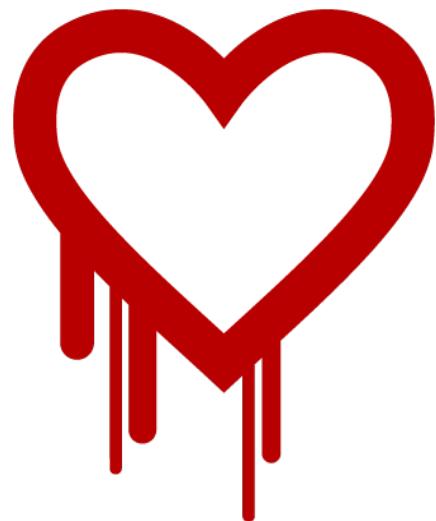
Privacy and Security Concerns







TARGET



U B E R



TARGET

Can we limit
third-party exposure?



Cryptography!

Example:

Multi-Device File Sync

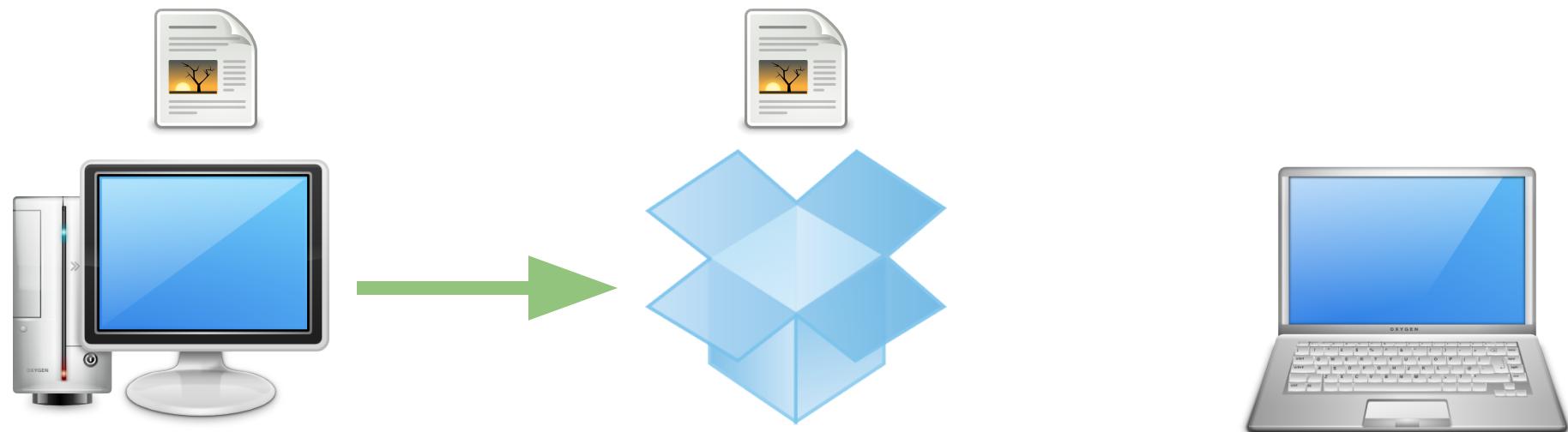
Multi-Device File Sync



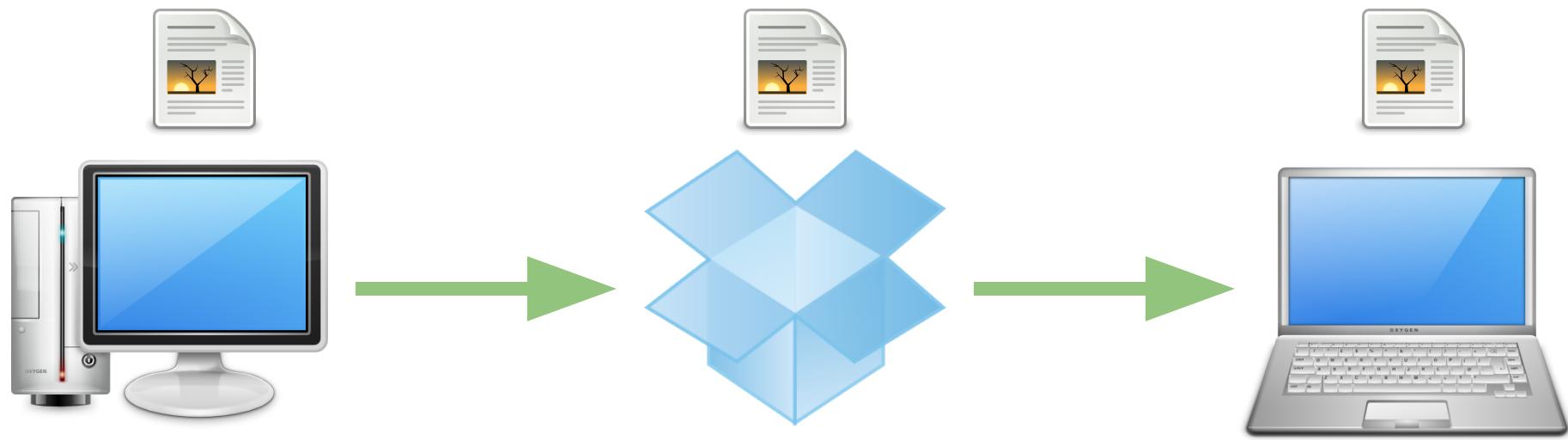
Multi-Device File Sync



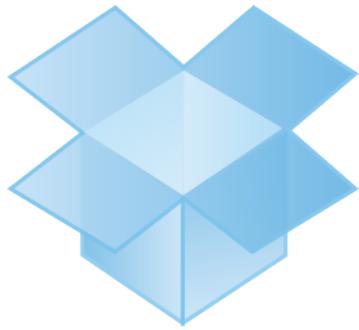
Multi-Device File Sync



Multi-Device File Sync

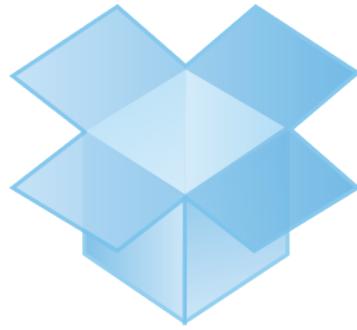
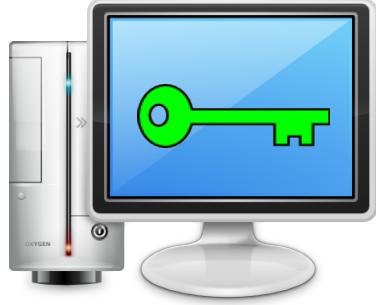


Multi-Device File Sync



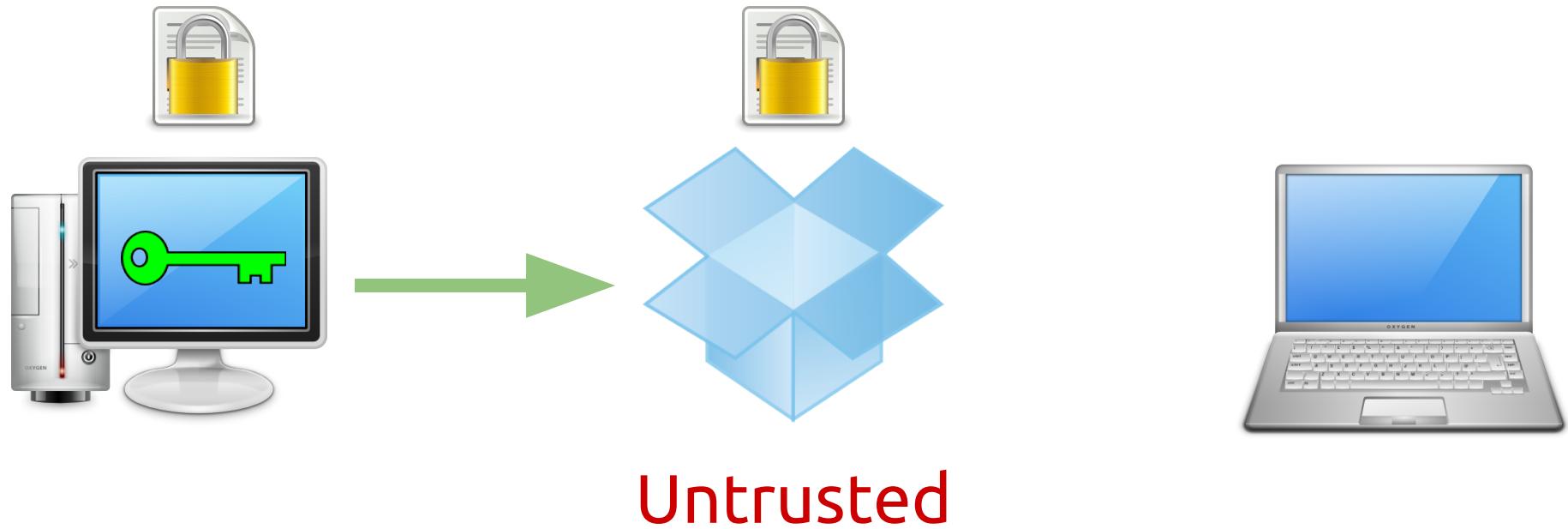
Untrusted

Multi-Device File Sync + Encryption

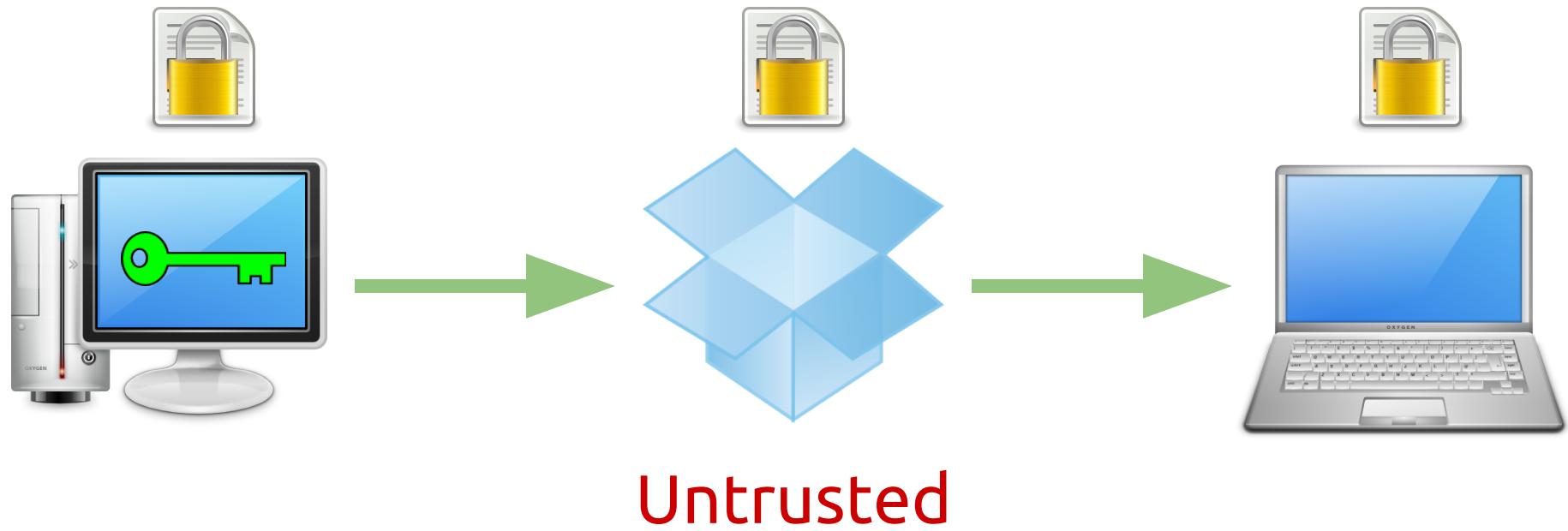


Untrusted

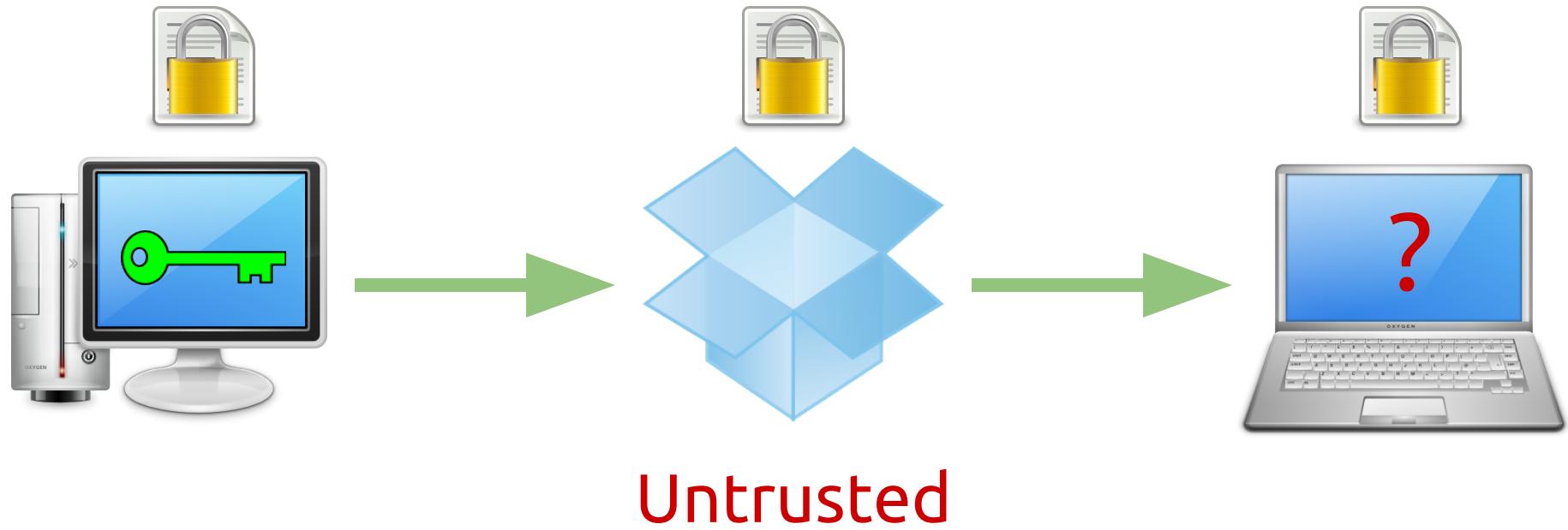
Multi-Device File Sync + Encryption



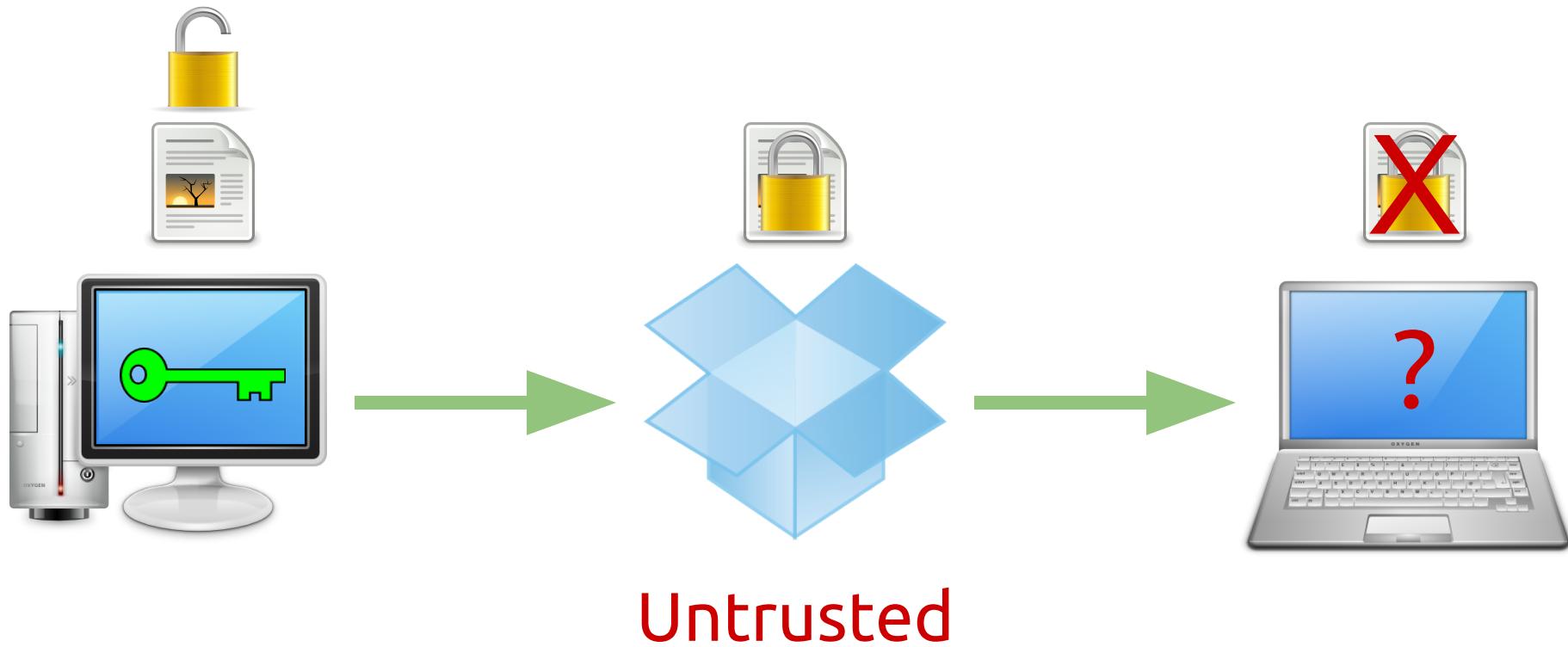
Multi-Device File Sync + Encryption



Multi-Device File Sync + Encryption



~~Multi-Device~~ File Sync + Encryption



Failure of Existing Solutions

Need for new solutions...

Key Management Solutions

Secret Storage Solutions

Related Work

Trust, Threat, and Security Modeling

Trust, Threat, and Security Modeling

Ali Abbas, et al. *A State of the Art Security Taxonomy of Internet Security: Threats and Countermeasures.* Computer Science and Engineering, 1(1):27–36, 2005.

Trust, Threat, and Security Modeling

Ali Abbas, et al. *A State of the Art Security Taxonomy of Internet Security: Threats and Countermeasures.* Computer Science and Engineering, 1(1):27–36, 2005.

K. Tsipenyuk, B. Chess, and G. McGraw. *Seven Pernicious Kingdoms: A Taxonomy of Software Security Errors.* IEEE Security & Privacy, 3(6):81–84, 2005.

Trust, Threat, and Security Modeling

Ali Abbas, et al. *A State of the Art Security Taxonomy of Internet Security: Threats and Countermeasures.* Computer Science and Engineering, 1(1):27–36, 2005.

K. Tsipenyuk, B. Chess, and G. McGraw. *Seven Pernicious Kingdoms: A Taxonomy of Software Security Errors.* IEEE Security & Privacy, 3(6):81–84, 2005.

Flowerday and Von Solms. *Trust: An Element of Information Security.* Security and Privacy in Dynamic Environments, 201: 87–98. Kluwer Academic Publishers, Boston, 2006.

Minimizing Third Party Trust

Minimizing Third Party Trust

Vipul Goyal, et al. *Attribute-based encryption for fine-grained access control of encrypted data*. CCS 06, New York, New York, USA, 2006. ACM Press.

Minimizing Third Party Trust

Vipul Goyal, et al. *Attribute-based encryption for fine-grained access control of encrypted data*. CCS 06, New York, New York, USA, 2006. ACM Press.

Craig Gentry. A Fully Homomorphic Encryption Scheme. PhD thesis, Stanford University, 2009.

Raluca Ada Popa, et al. *CryptDB: Protecting Confidentiality with Encrypted Query Processing*. SOSP '11, New York, New York, USA, 2011. ACM Press.

Minimizing Third Party Trust

Vipul Goyal, et al. *Attribute-based encryption for fine-grained access control of encrypted data*. CCS 06, New York, New York, USA, 2006. ACM Press.

Craig Gentry. A Fully Homomorphic Encryption Scheme. PhD thesis, Stanford University, 2009.

Raluca Ada Popa, et al. *CryptDB: Protecting Confidentiality with Encrypted Query Processing*. SOSP '11, New York, New York, USA, 2011. ACM Press.

Zooko Wilcox-O'Hearn and Brian Warner. *Tahoe: The Least-Authority Filesystem*. Workshop on Storage Security and Survivability, New York, New York, USA, 2008. ACM Press.

Enhancing End-User Security

Enhancing End-User Security

Communication Tools

PGP Revisited, TextSecure, OTR Protocol, Etc

Enhancing End-User Security

Communication Tools

PGP Revisited, TextSecure, OTR Protocol, Etc

Password Managers

LastPass, OnePassword, etc

Enhancing End-User Security

Communication Tools

PGP Revisited, TextSecure, OTR Protocol, Etc

Password Managers

LastPass, OnePassword, etc

Secure Storage

SpiderOak, BitTorrent Sync, Least Authority (Tahoe LAFS)

Key Management

Key Management

David Mazieres, et al. *Separating Key Management from File System Security*. ACM SIGOPS Operating Systems Review, 33(5):124–139, December 1999.

Key Management

David Mazieres, et al. *Separating Key Management from File System Security*. ACM SIGOPS Operating Systems Review, 33(5):124–139, December 1999.

Matt Blaze. *Oblivious Key Escrow*. Information Hiding, 1996.

Key Management

David Mazieres, et al. *Separating Key Management from File System Security*. ACM SIGOPS Operating Systems Review, 33(5):124–139, December 1999.

Matt Blaze. *Oblivious Key Escrow*. Information Hiding, 1996.

Cloud Key Management

Cloudkeep/Barbican, Amazon CloudHSM, Gazzang zTrustee

Analyzing Trust

Who are we trusting and with what capabilities?

Who are we trusting and with what capabilities?

How can such trust be violated?

Who are we trusting and with what capabilities?

How can such trust be violated?

What is the effect of violating such trust?

Analysis Framework

Analysis Framework

Degree of Trust
(Capabilities)

Analysis Framework

Degree of Trust
(Capabilities)

Types of Violation
(Attacks)

Degree of Trust

Storage (S)

Access (R)

Manipulation (W)

Meta-Analysis (M)

Degree of Trust

Storage (S)

*Can a third party faithfully store private user data
and make it available to the user upon request?*

Access (R)

Manipulation (W)

Meta-Analysis (M)

Degree of Trust

Storage (S)

Access (R)

*Can a third party read and interpret
the private user data they store?*

Manipulation (W)

Meta-Analysis (M)

Degree of Trust

Storage (S)

Access (R)

Manipulation (W)

*Can a third party modify the
private user data to which they have access?*

Meta-Analysis (M)

Degree of Trust

Storage (S)

Access (R)

Manipulation (W)

Meta-Analysis (M)

*Can a third party gather user metadata
related to any stored private user data?*

Types of Violation

Implicit (P)

Compelled (C)

Unintentional (U)

Insider (I)

Outsider (O)

Types of Violation

Implicit (P)

*Occurs when a third party violates a user's trust
in a manner approved by the third party.*

Compelled (C)

Unintentional (U)

Insider (I)

Outsider (O)

Types of Violation

Implicit (P)
Compelled (C)

*Occurs when a third party is compelled
by another actor to violate a user's trust.*

Unintentional (U)
Insider (I)
Outsider (O)

Types of Violation

Implicit (P)

Compelled (C)

Unintentional (U)

*Occurs when a third party
unintentionally violates a user's trust.*

Insider (I)

Outsider (O)

Types of Violation

Implicit (P)

Compelled (C)

Unintentional (U)

Insider (I)

Occurs when a privileged adversary within the third party independently violates a user's trust.

Outsider (O)

Types of Violation

Implicit (P)

Compelled (C)

Unintentional (U)

Insider (I)

Outsider (O)

Occurs when an external adversary gains unauthorized access to private user data stored by third party.

Degree of Trust

Storage (S)
Access (R)
Manipulation (W)
Meta-Analysis (M)

Types of Violation

Implicit (P)
Compelled (C)
Unintentional (U)
Insider (I)
Outsider (O)

Degree of Trust

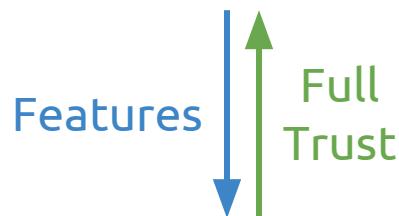
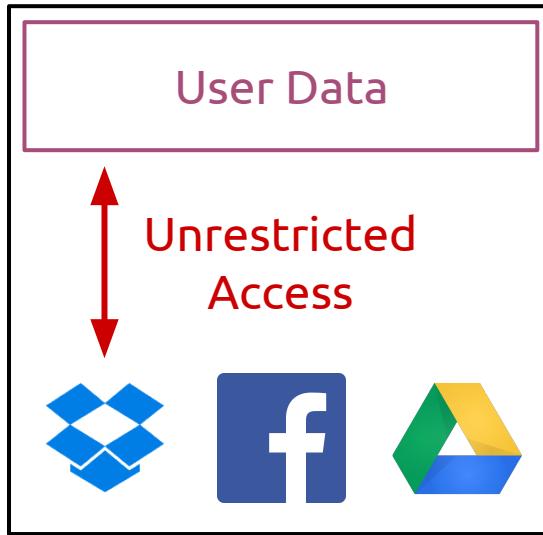
Storage (S)
Access (R)
Manipulation (W)
Meta-Analysis (M)

Types of Violation

Implicit (P)
Compelled (C)
Unintentional (U)
Insider (I)
Outsider (O)

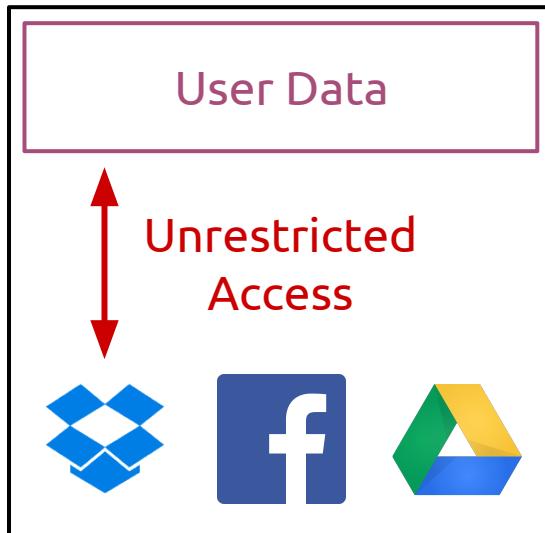
Traditional Trust Model

Feature Provider

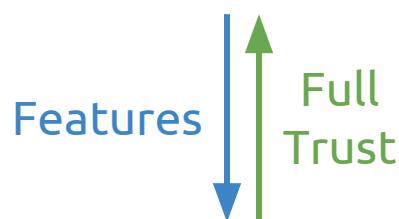


Traditional Trust Model

Feature Provider

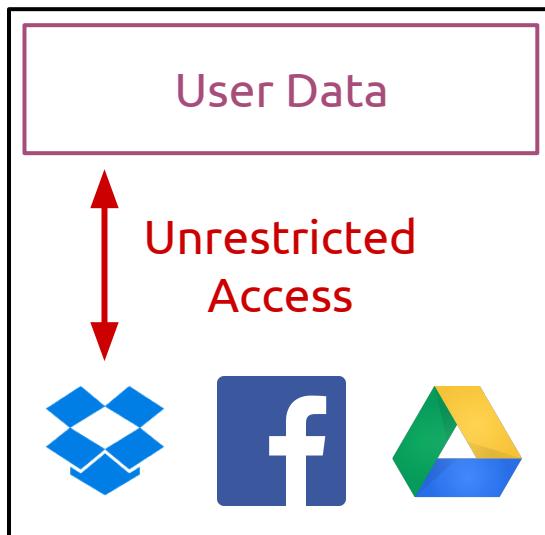


Storage (S)

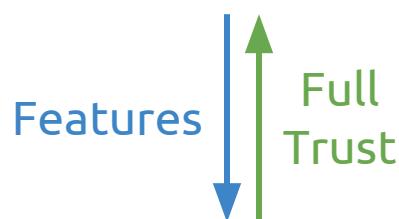


Traditional Trust Model

Feature Provider

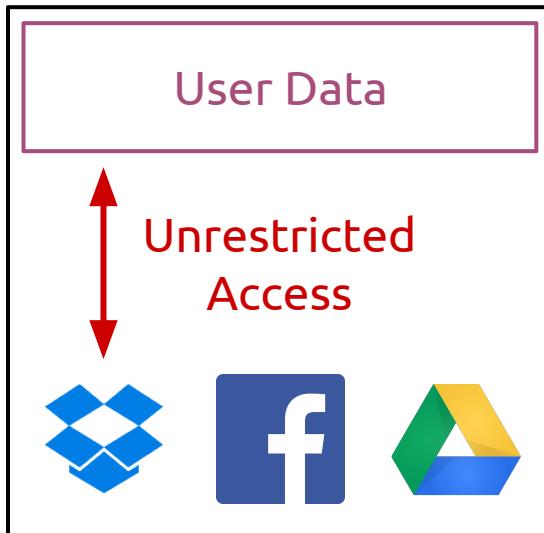


Storage (S)
Access (R)

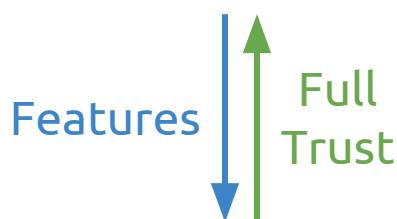


Traditional Trust Model

Feature Provider

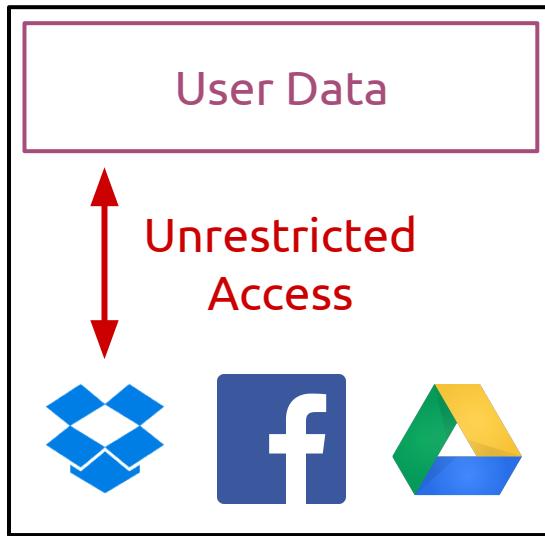


Storage (S)
Access (R)
Manipulation (W)

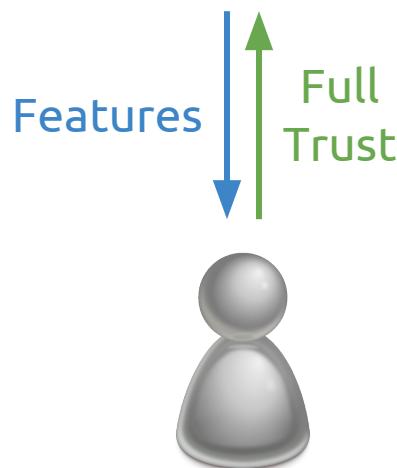


Traditional Trust Model

Feature Provider

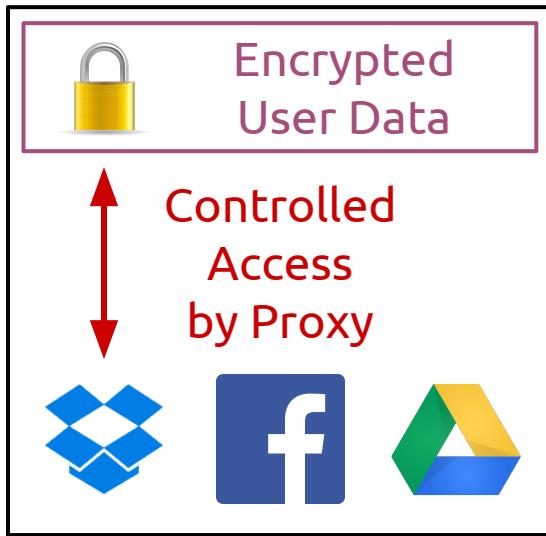


Storage (S)
Access (R)
Manipulation (W)
Meta-Analysis (M)

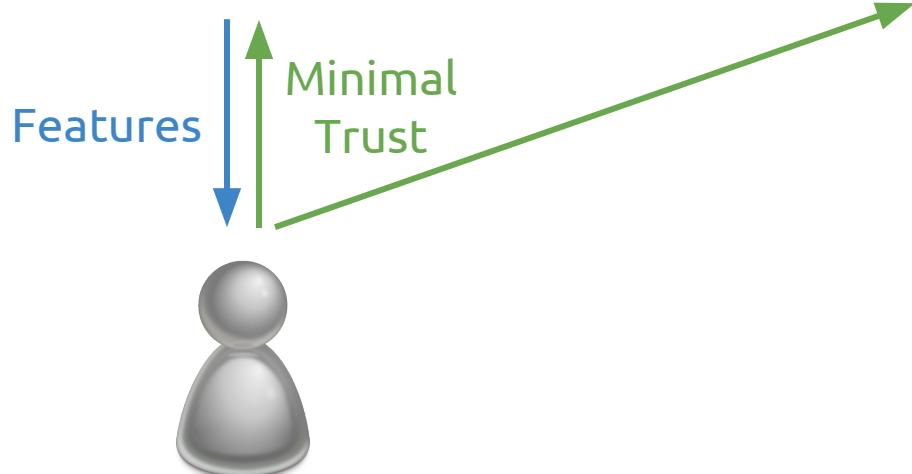


SSaaS Trust Model

Feature Provider

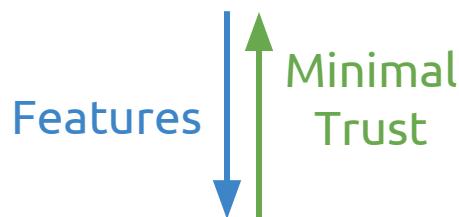
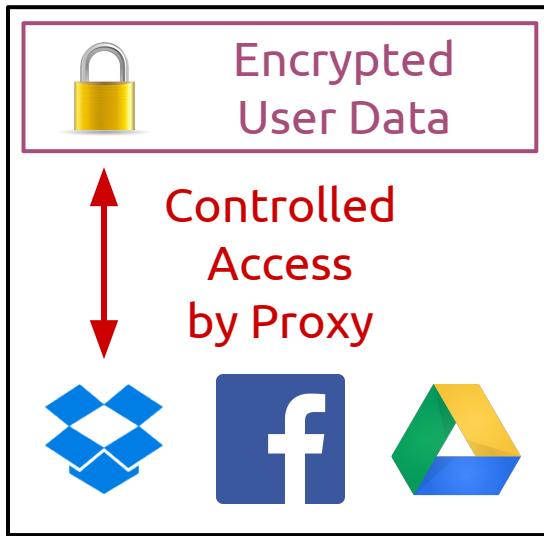


Secret Storage Provider



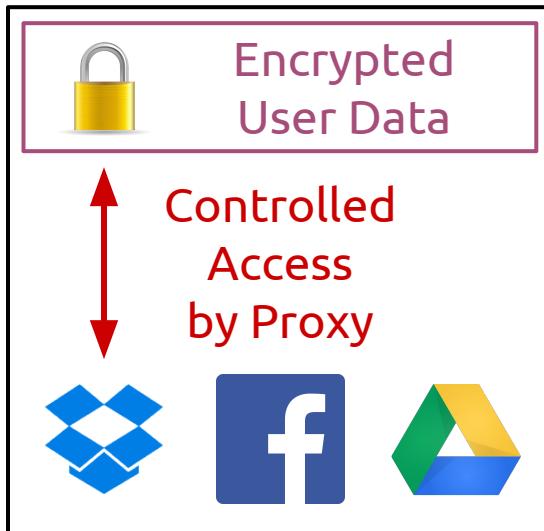
SSaaS Trust Model - FP

Feature Provider

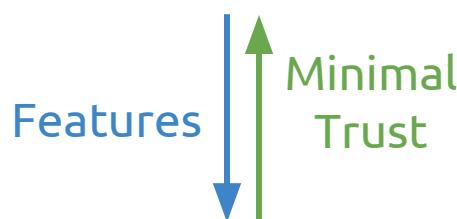


SSaaS Trust Model - FP

Feature Provider

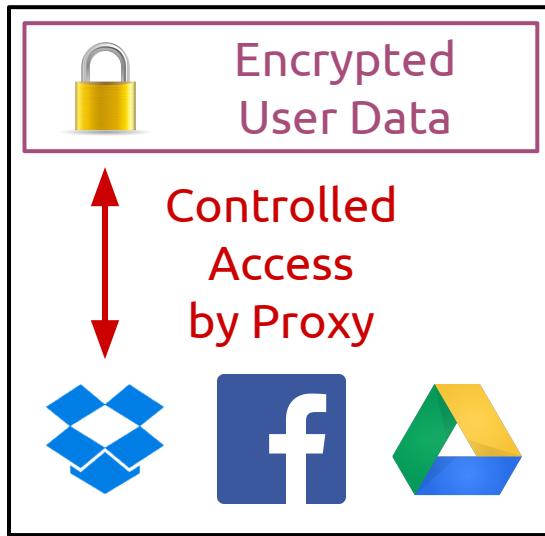


Storage (S)

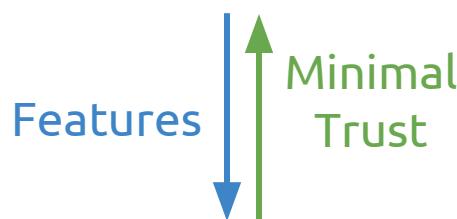


SSaaS Trust Model - FP

Feature Provider

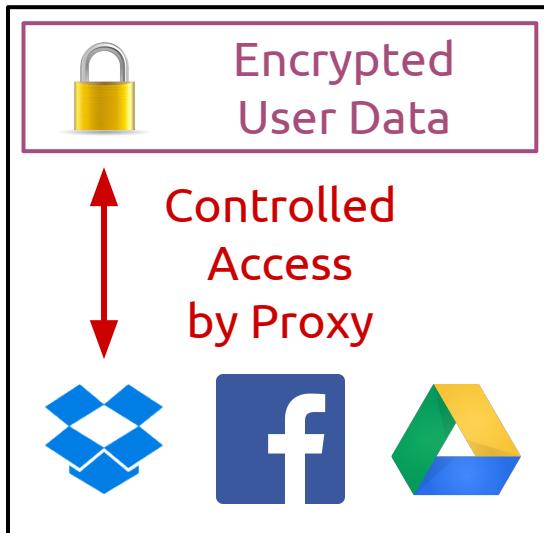


Storage (S)
Access (R)

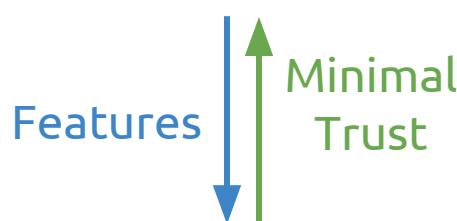


SSaaS Trust Model - FP

Feature Provider

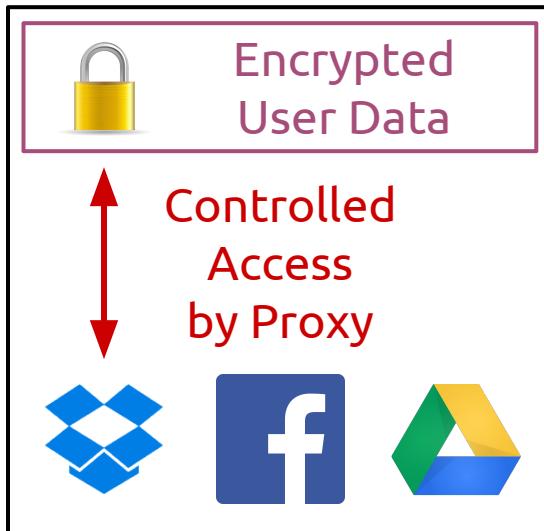


Storage (S)
~~Access (R)~~

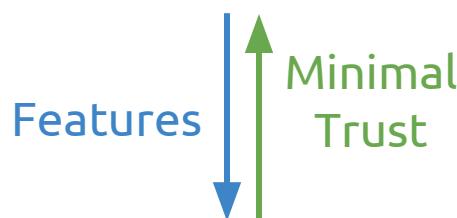


SSaaS Trust Model - FP

Feature Provider

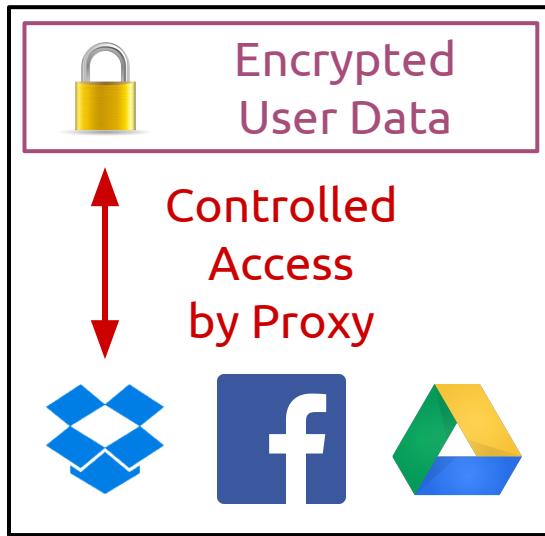


Storage (S)
~~Access (R)~~
Manipulation (W)

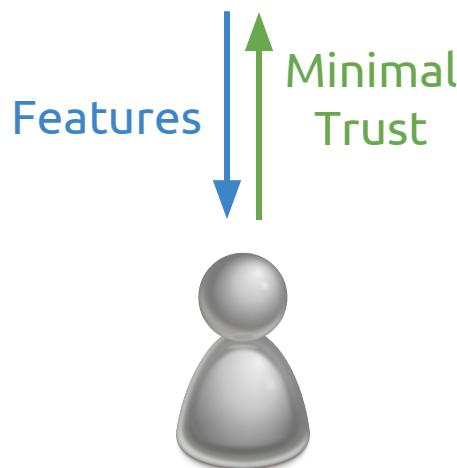


SSaaS Trust Model - FP

Feature Provider

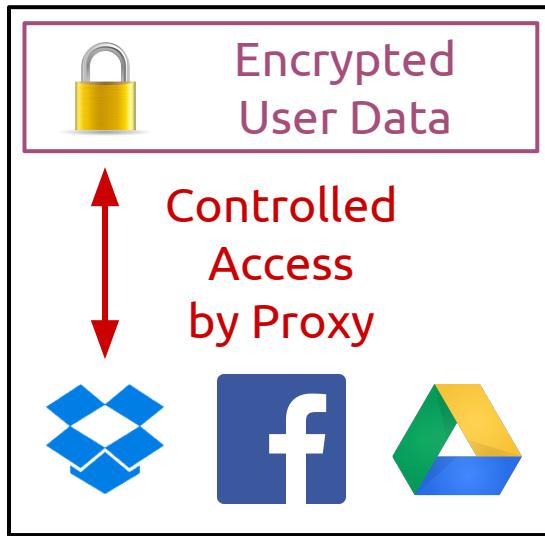


Storage (S)
~~Access (R)~~
~~Manipulation (W)~~

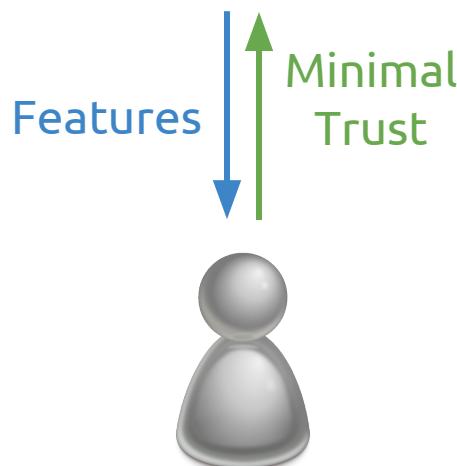


SSaaS Trust Model - FP

Feature Provider

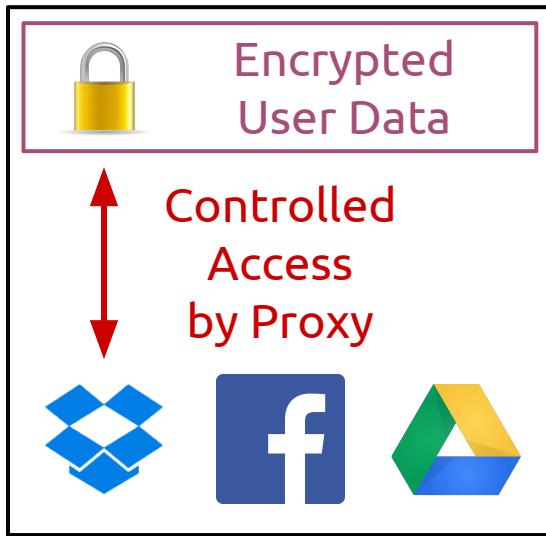


Storage (S)
~~Access (R)~~
~~Manipulation (W)~~
Meta-Analysis (M)

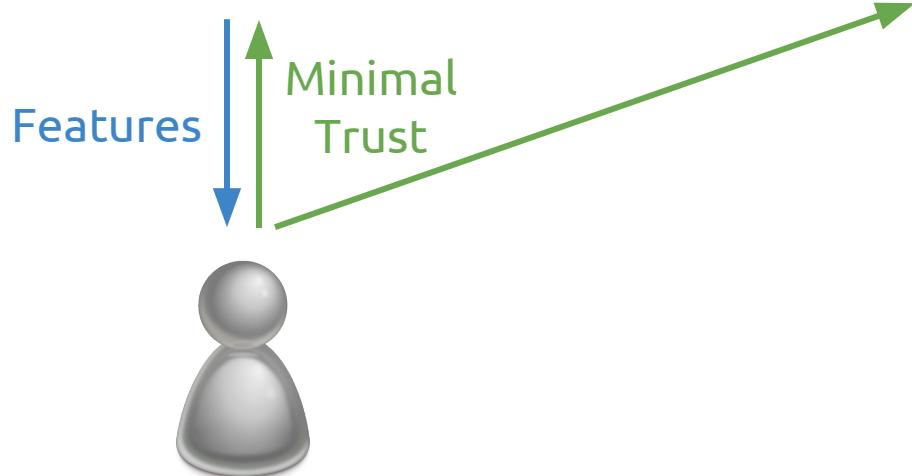


SSaaS Trust Model

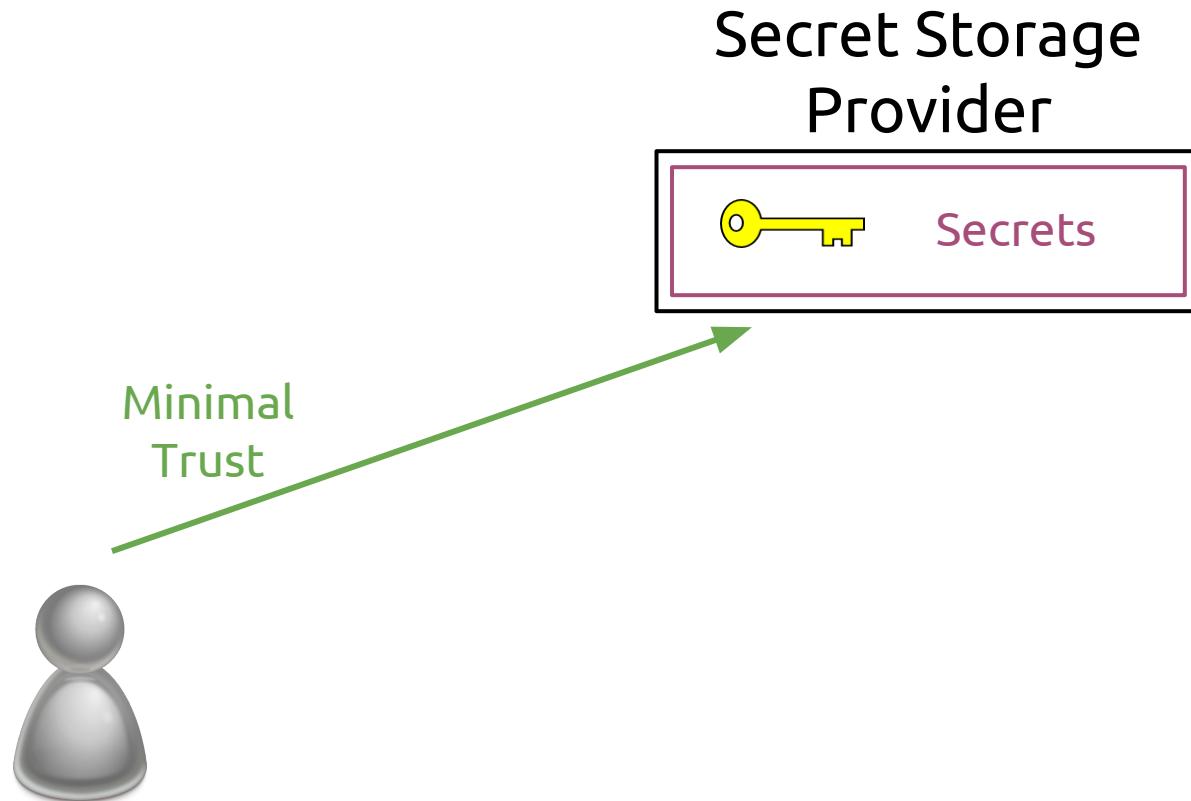
Feature Provider



Secret Storage Provider



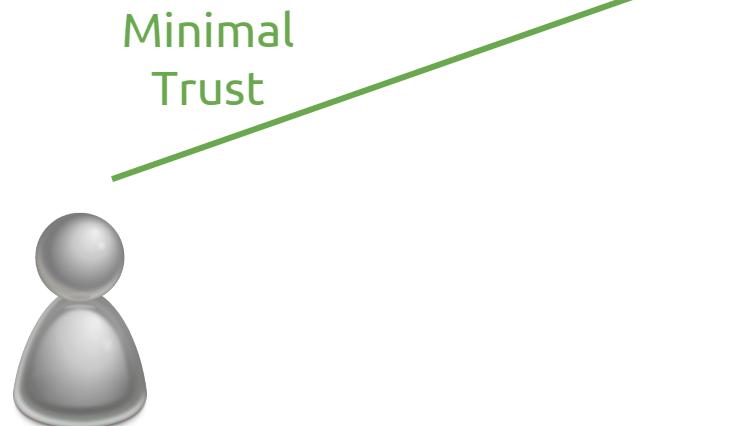
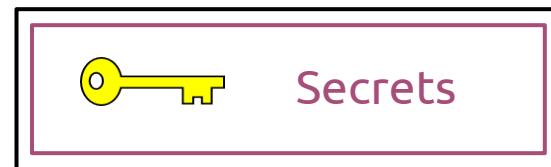
SSaaS Trust Model - SSP



SSaaS Trust Model - SSP

Storage (S)

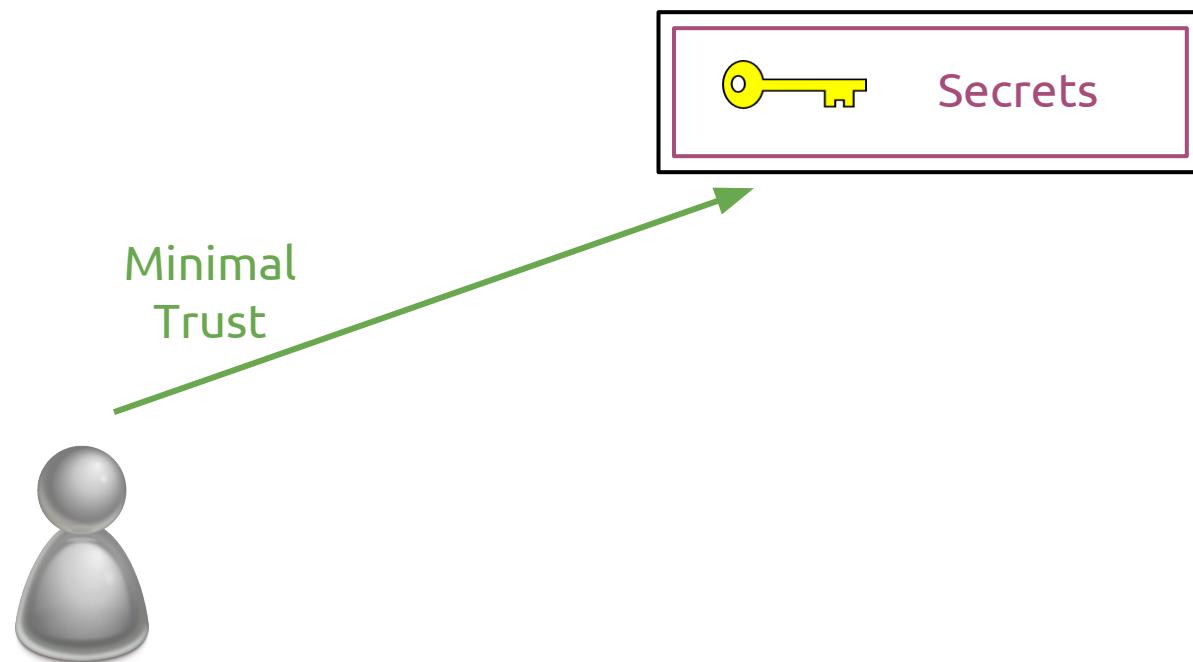
Secret Storage
Provider



SSaaS Trust Model - SSP

Storage (S)
Access (R)

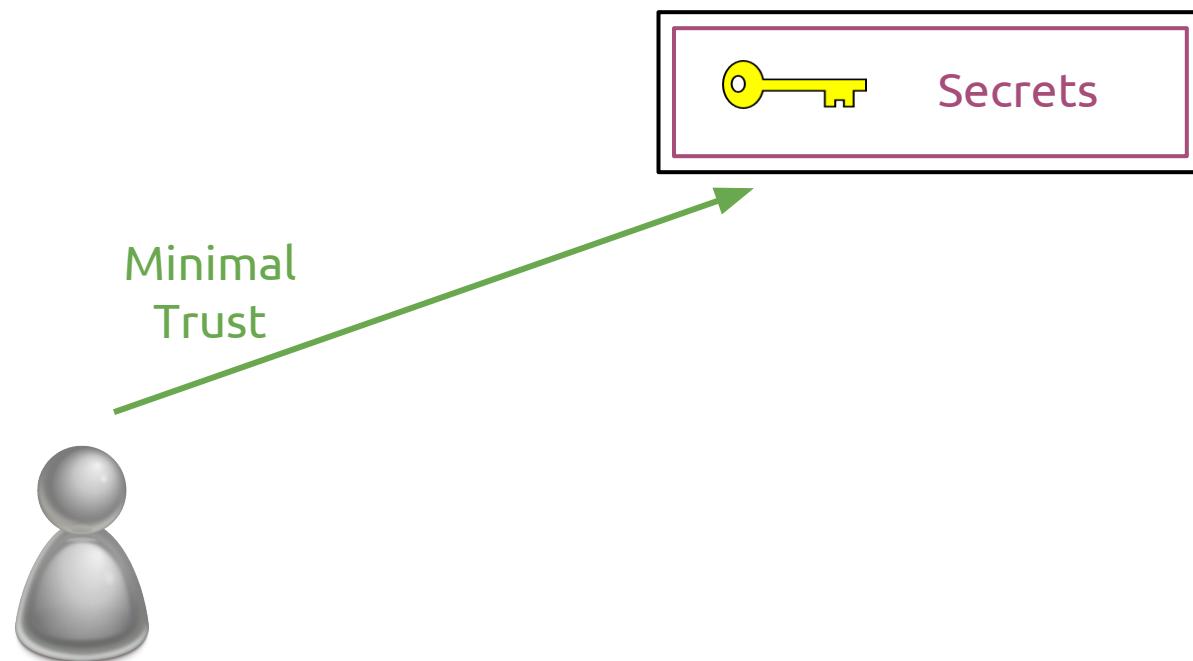
Secret Storage
Provider



SSaaS Trust Model - SSP

Storage (S)
~~Access (R)~~

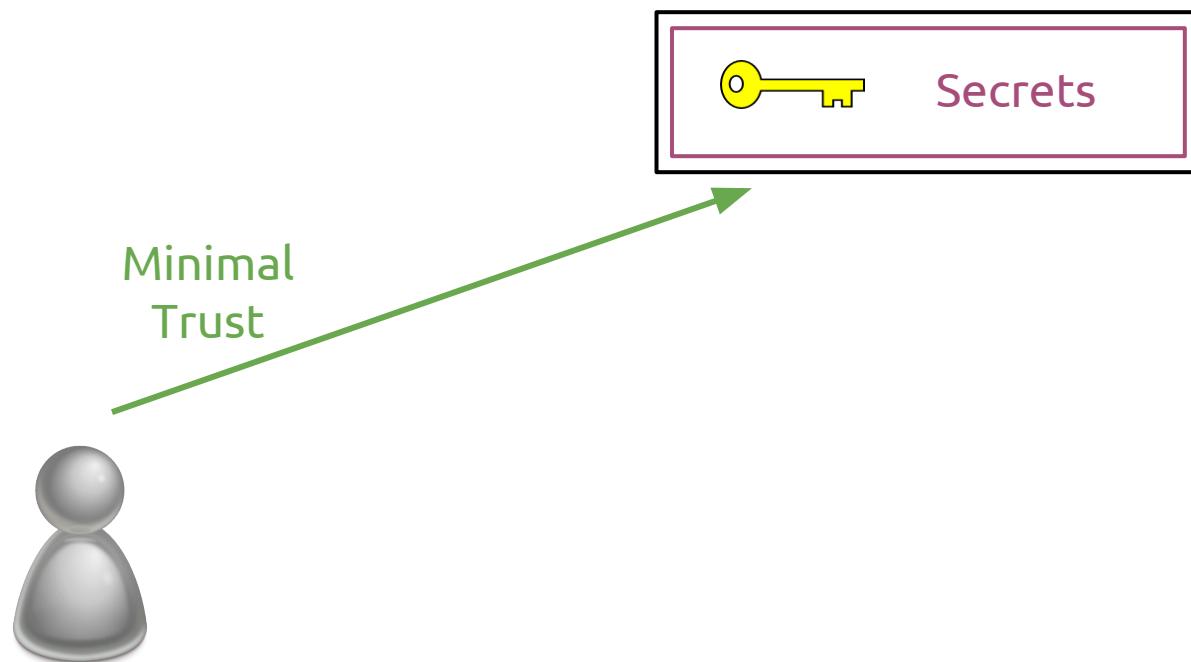
Secret Storage
Provider



SSaaS Trust Model - SSP

Storage (S)
~~Access (R)~~
Manipulation (W)

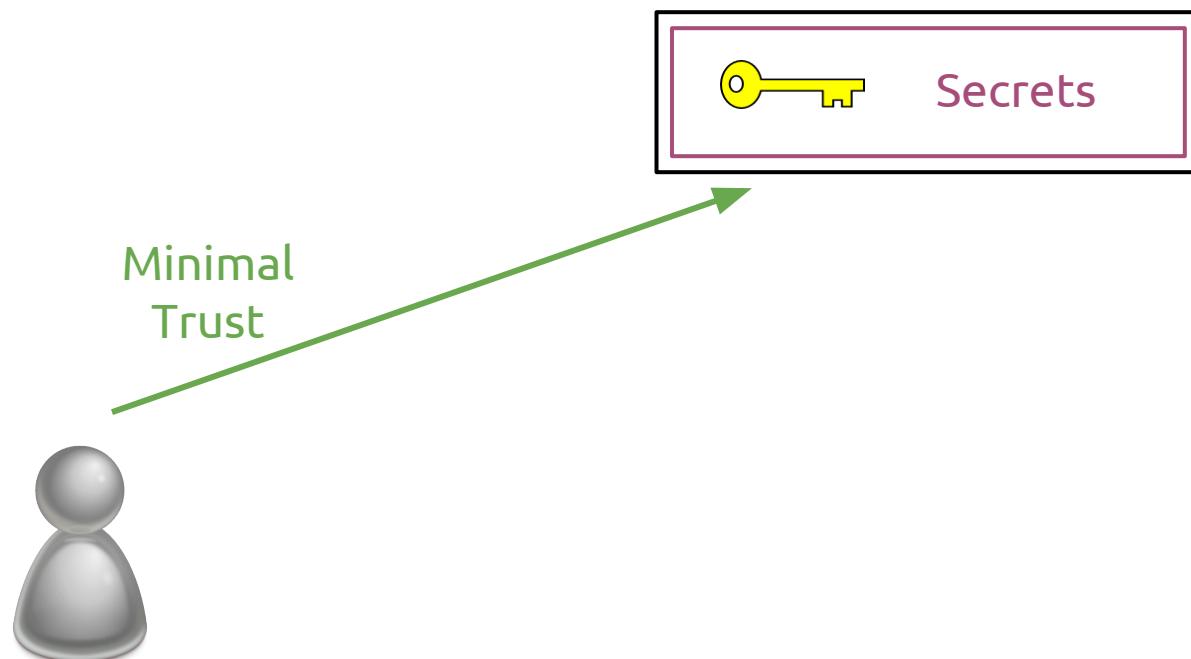
Secret Storage
Provider



SSaaS Trust Model - SSP

Storage (S)
~~Access (R)~~
~~Manipulation (W)~~

Secret Storage
Provider



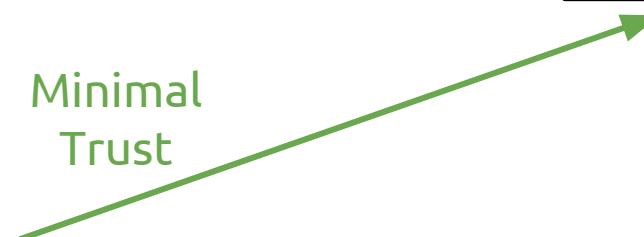
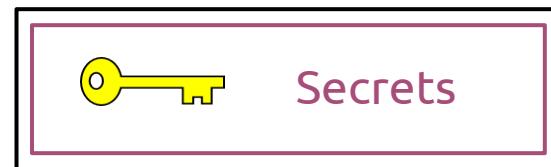
SSaaS Trust Model - SSP

Storage (S)
~~Access (R)~~
~~Manipulation (W)~~
Meta-Analysis (M)



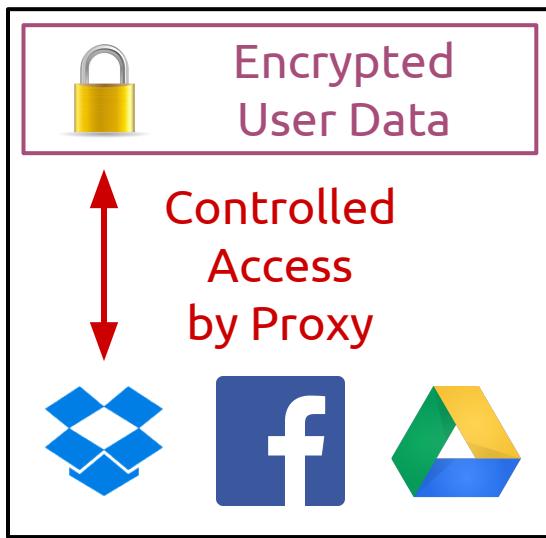
Minimal
Trust

Secret Storage
Provider

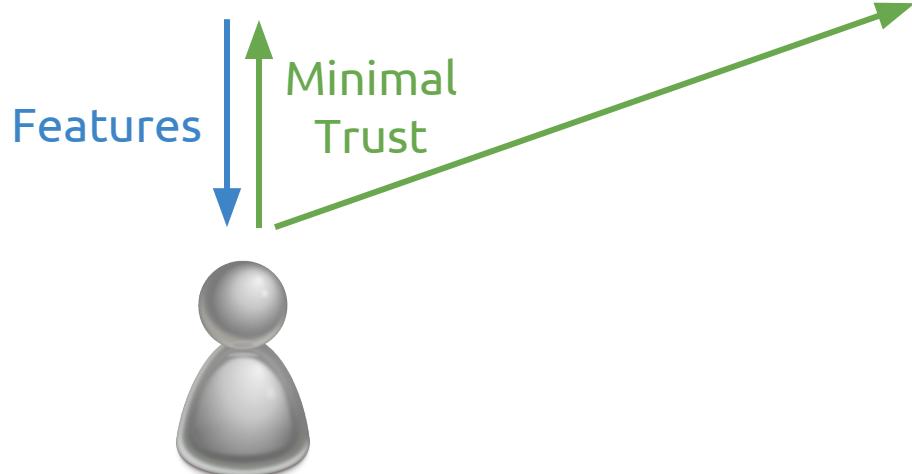


SSaaS Trust Model

Feature Provider

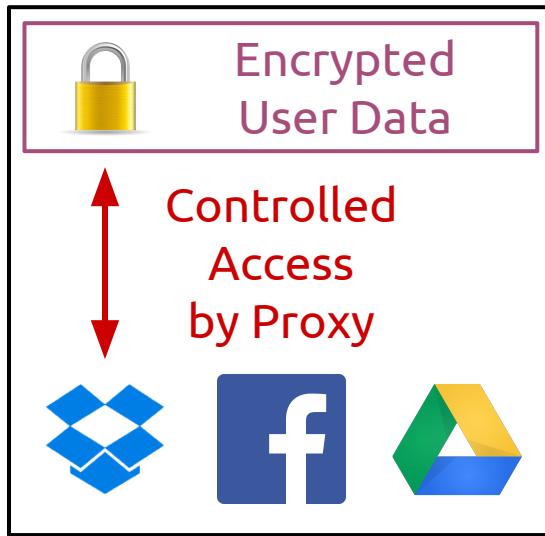


Secret Storage Provider

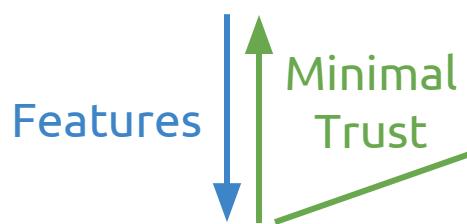


SSaaS Trust Model

Feature Provider

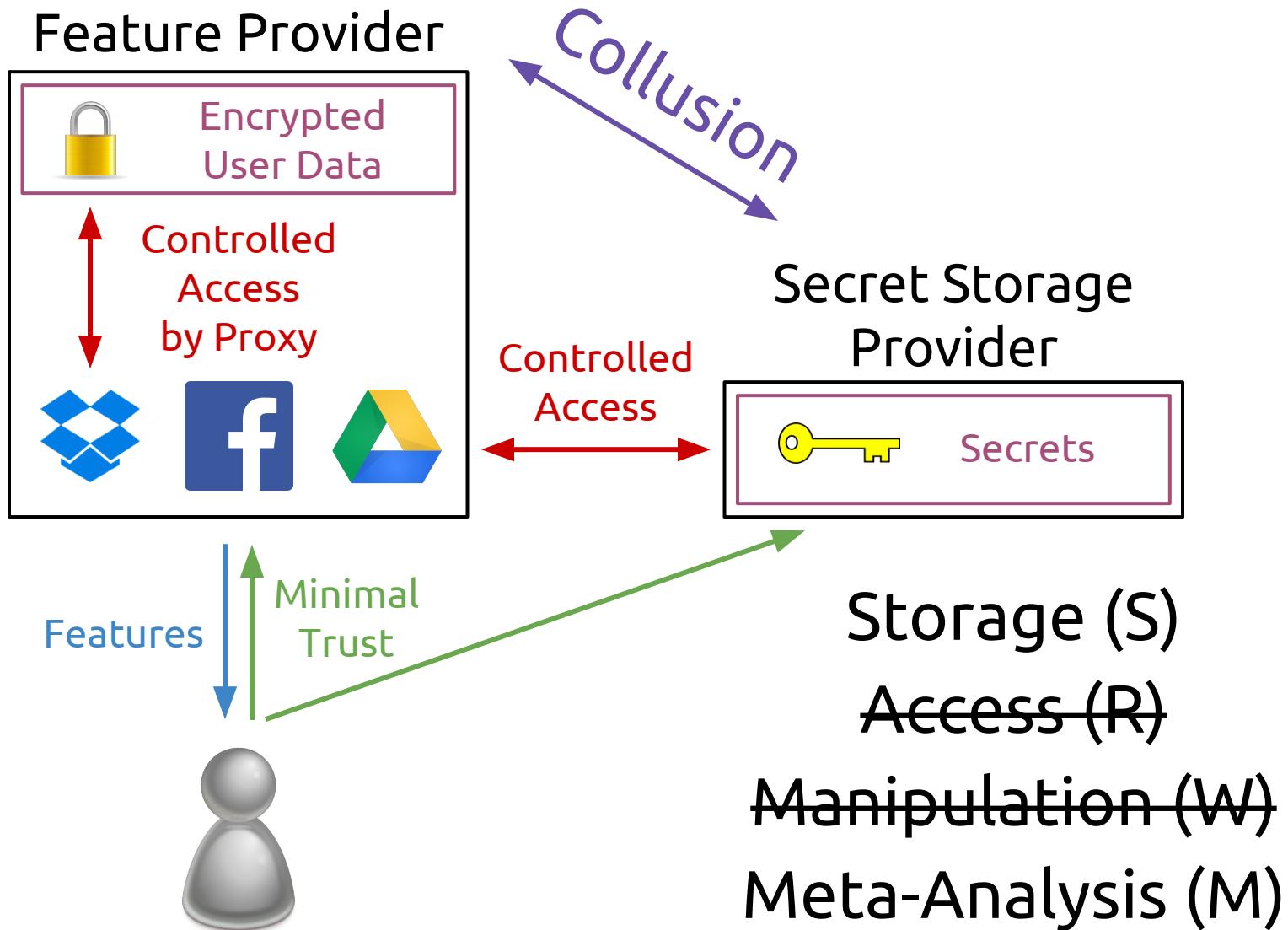


Secret Storage Provider

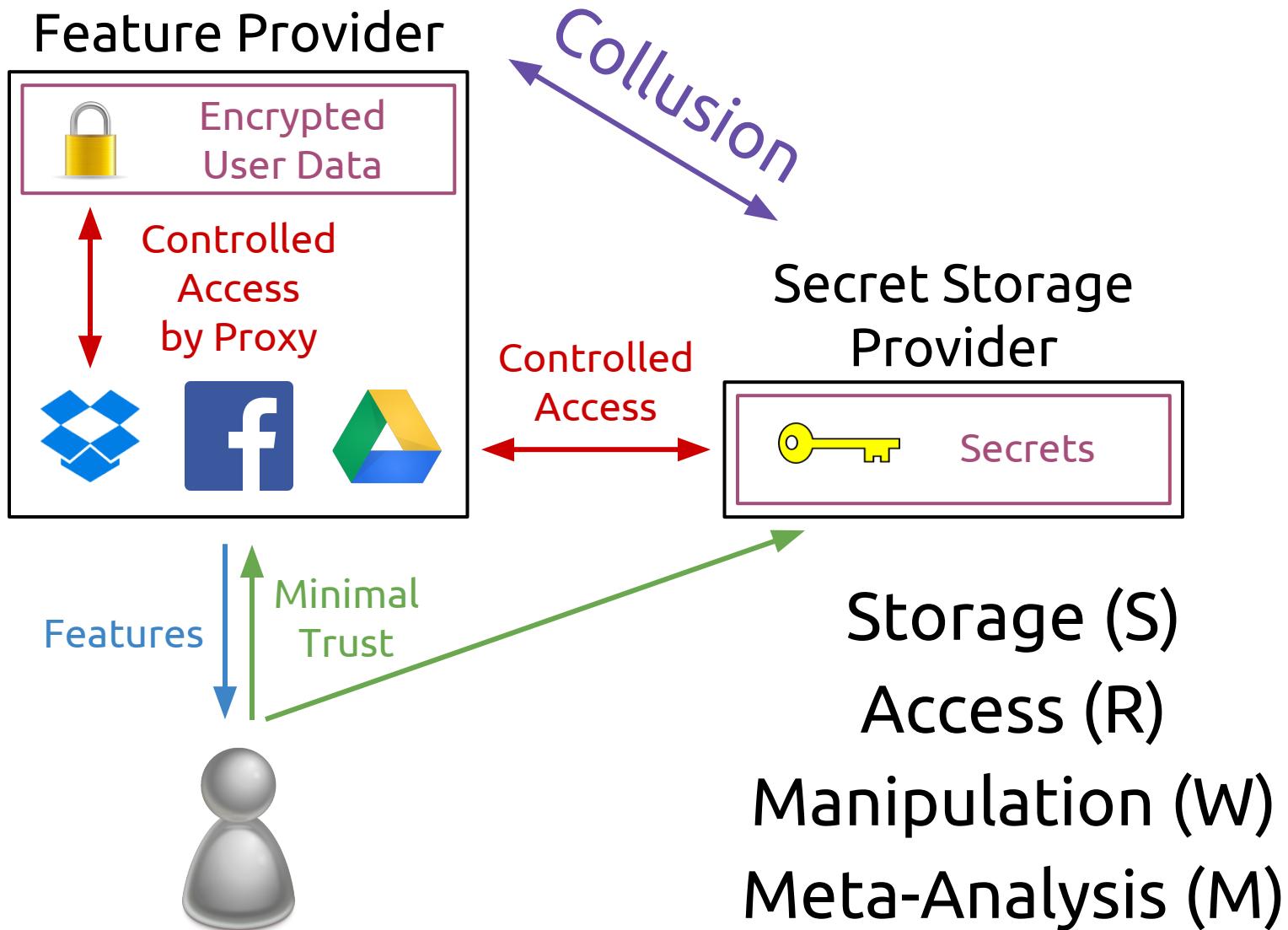


Storage (S)
~~Access (R)~~
~~Manipulation (W)~~
Meta-Analysis (M)

SSaaS Trust Model



SSaaS Trust Model



Types of Violation

Implicit (P)

Compelled (C)

Unintentional (U)

Insider (I)

Outsider (O)

Colluding (L)

Occurs when multiple trusted parties collude to gain capabilities beyond what the user intended each to have.

“Secret Storage as a Service” (SSaaS)

Core Features

Centralized Secret Storage

Centralized Secret Storage

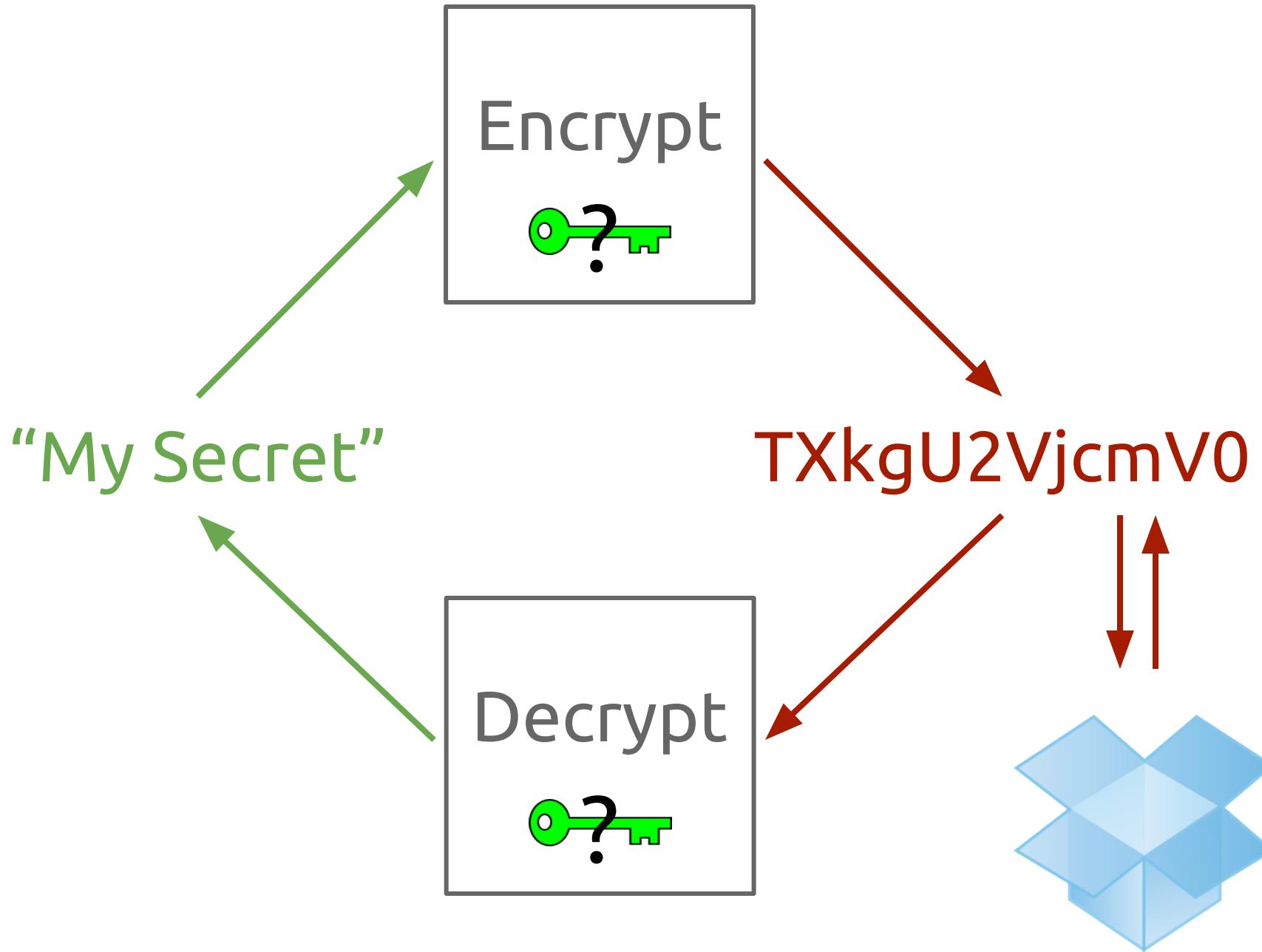
Flexible Access Control

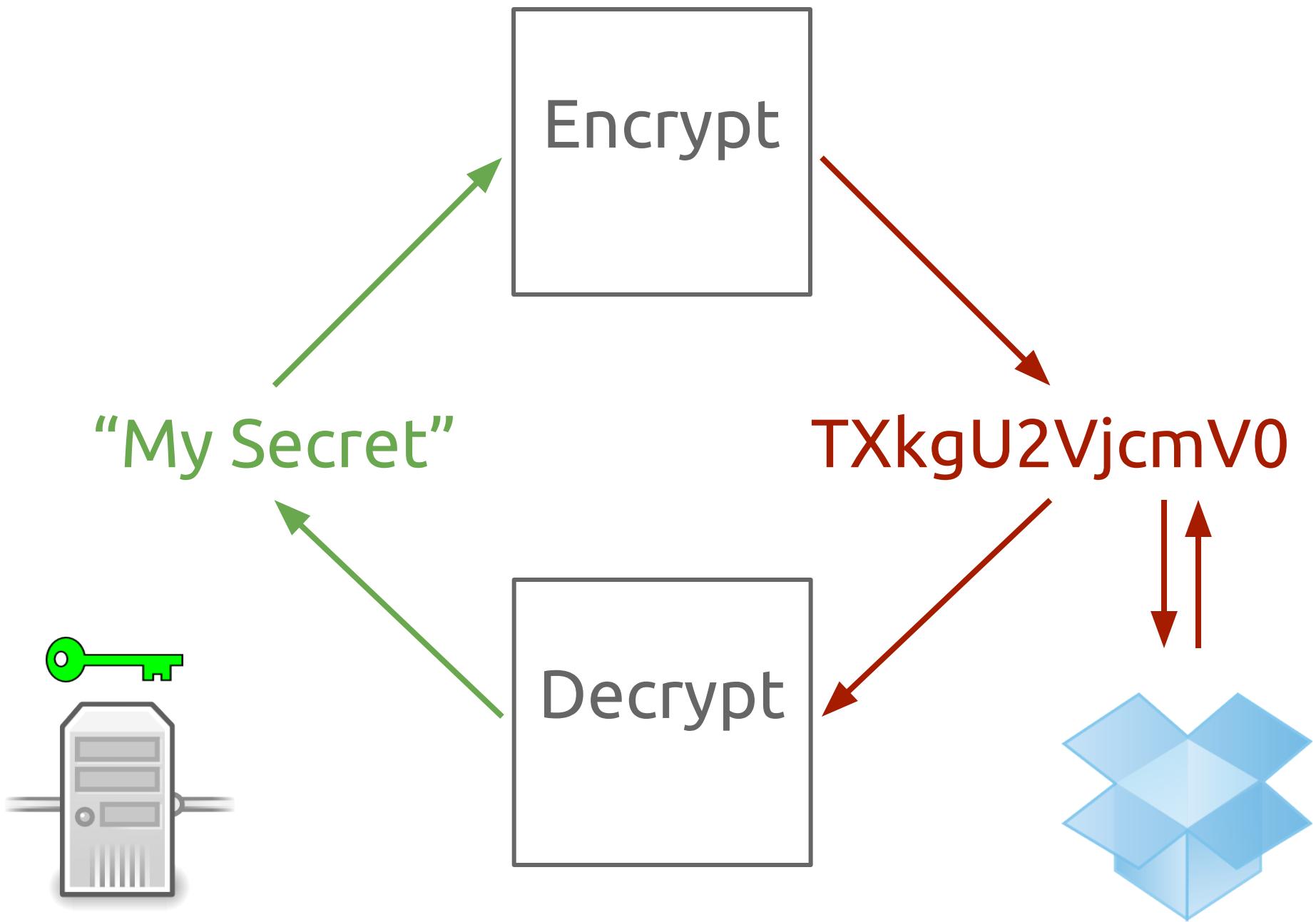
Centralized Secret Storage

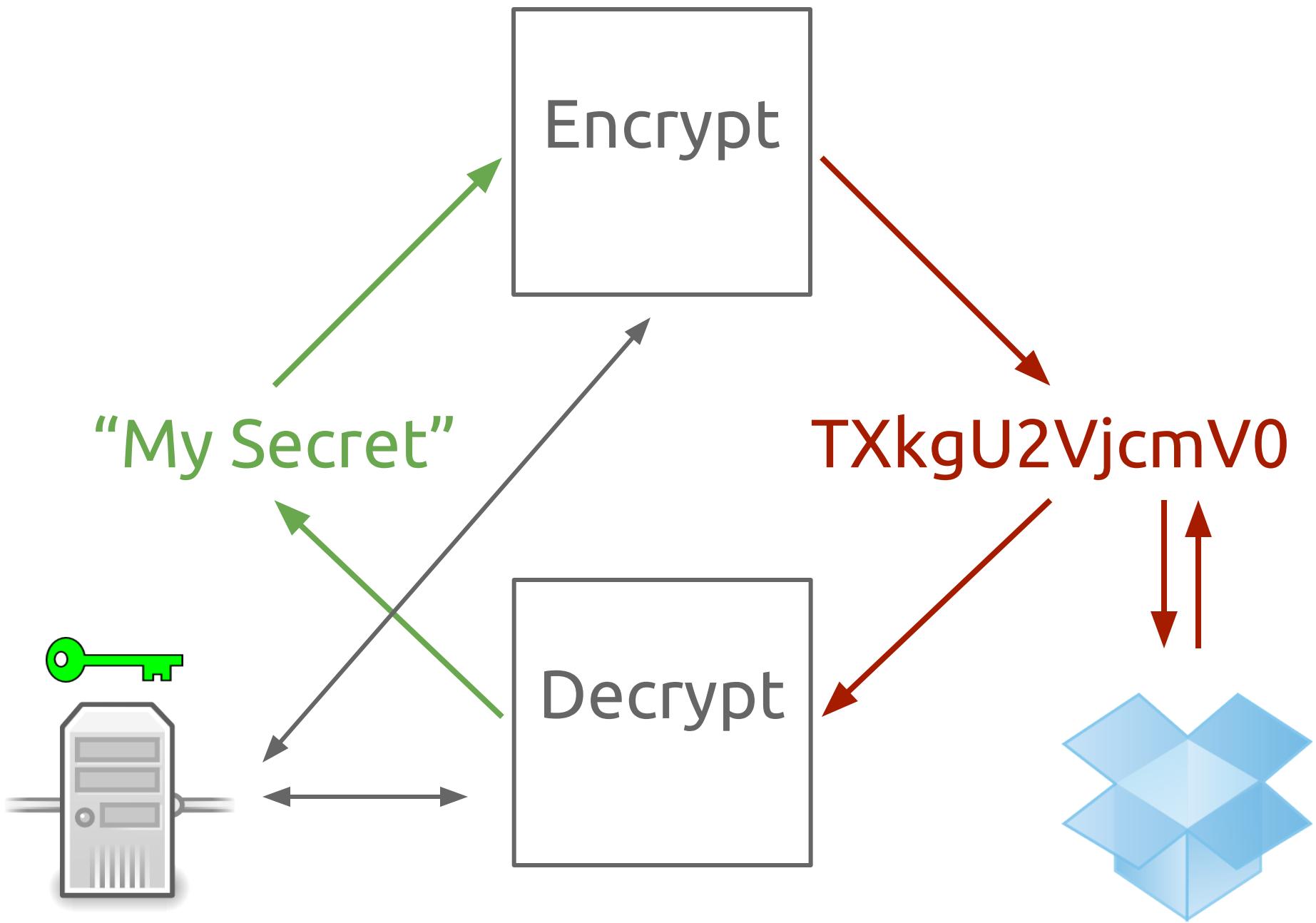
Flexible Access Control

Auditing and Revocation

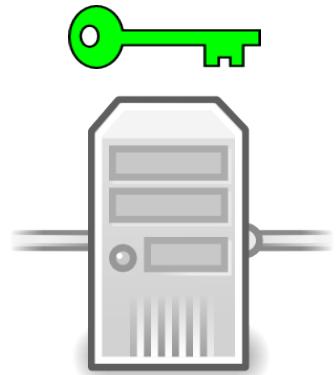
SSaaS Architecture







Secret Storage Provider (SSP)



Feature Provider (FP)



Stored Secrets

Stored Secrets

Passwords

Stored Secrets

Passwords

Personal Data

Stored Secrets

Passwords

Personal Data

Cryptographic Keys

Stored Secrets

Passwords

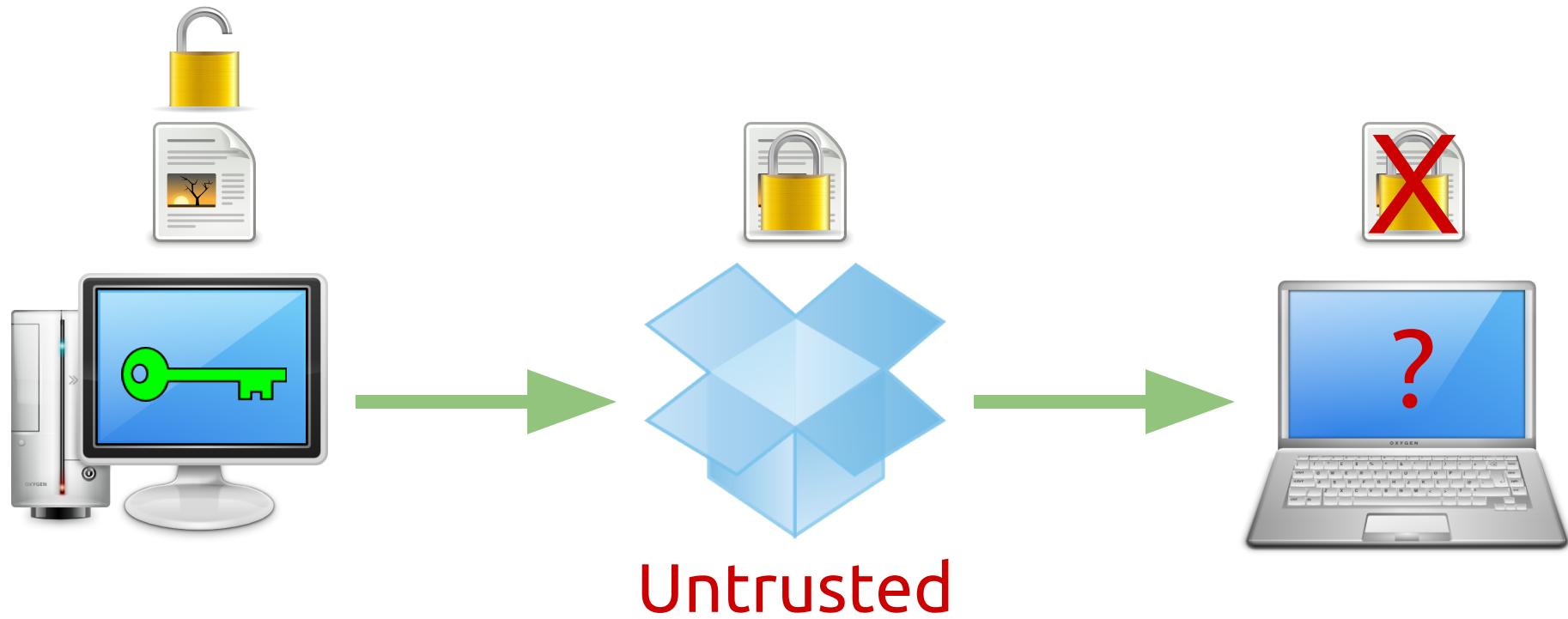
Personal Data

Cryptographic Keys

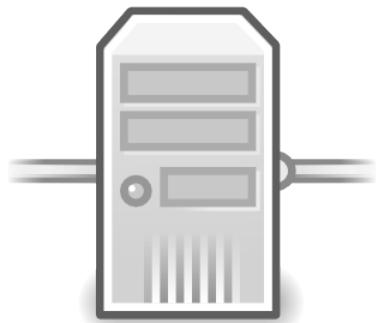
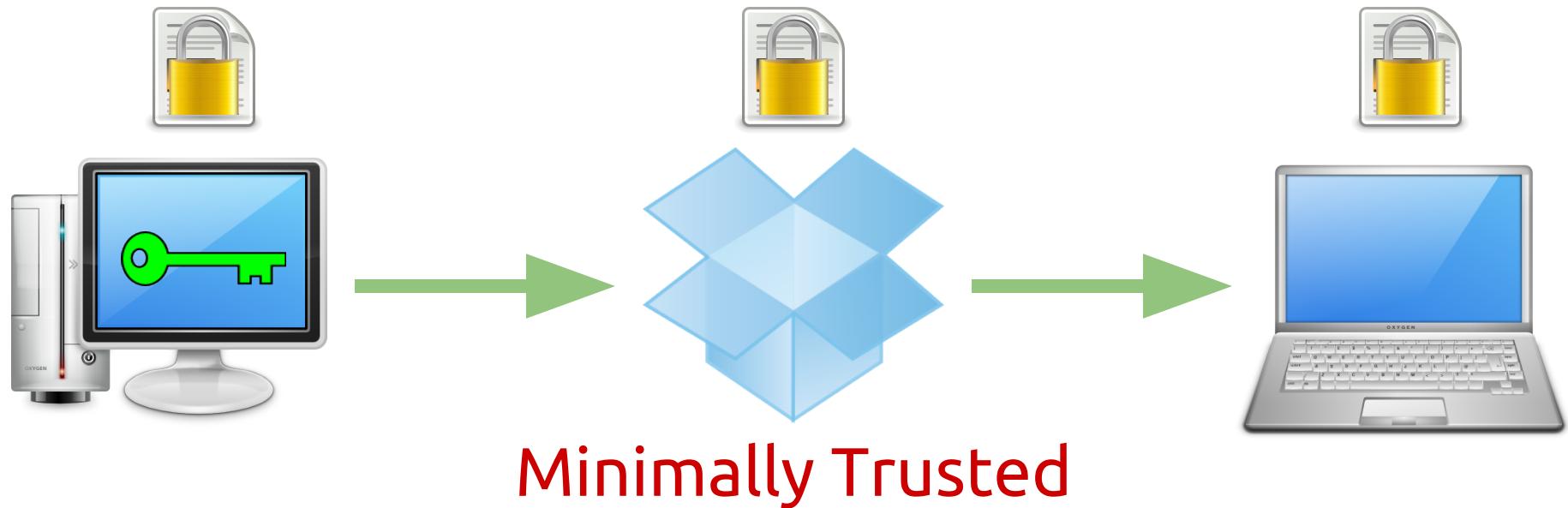
Etc...

Revisiting Example

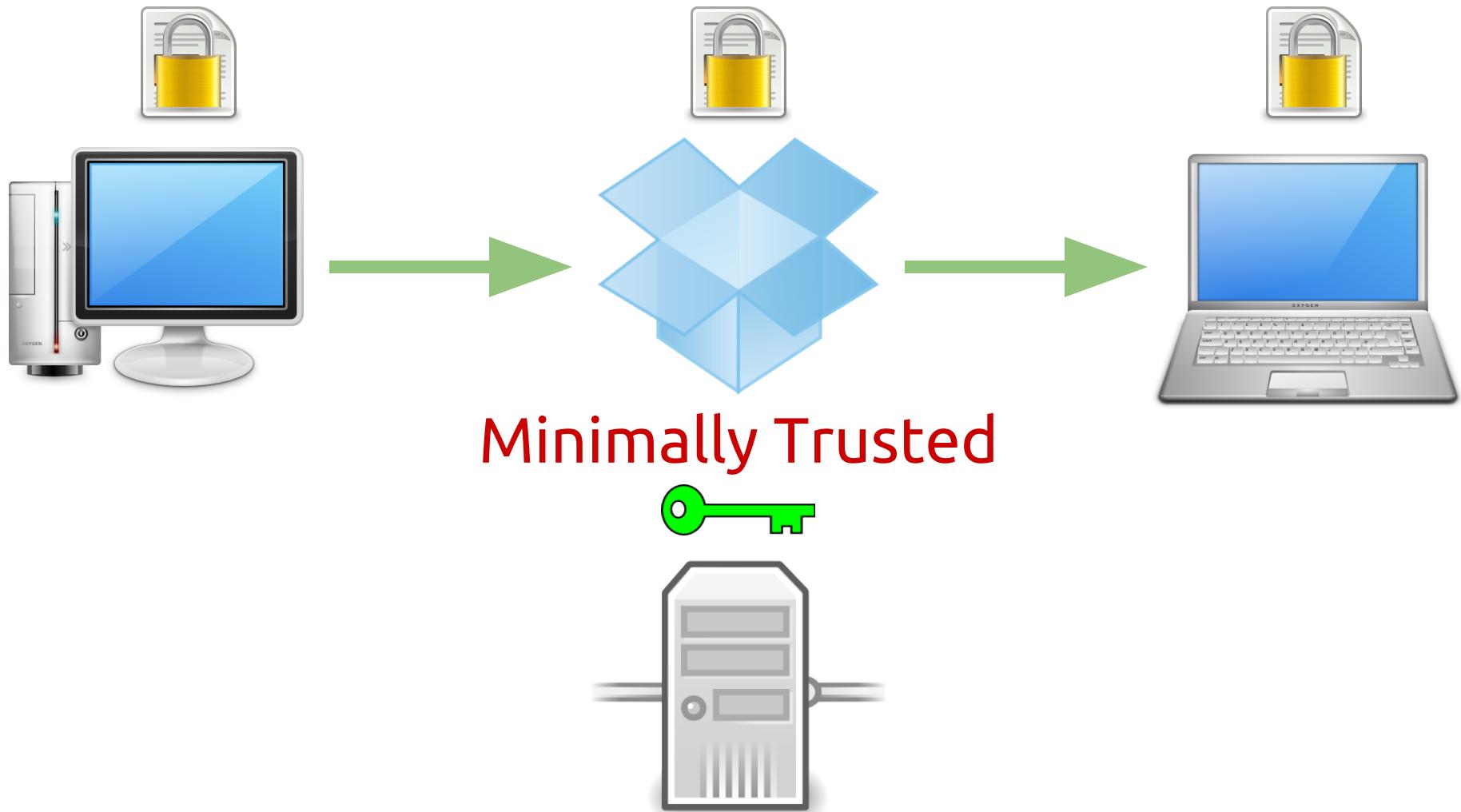
Multi-Device File Sync + Encryption



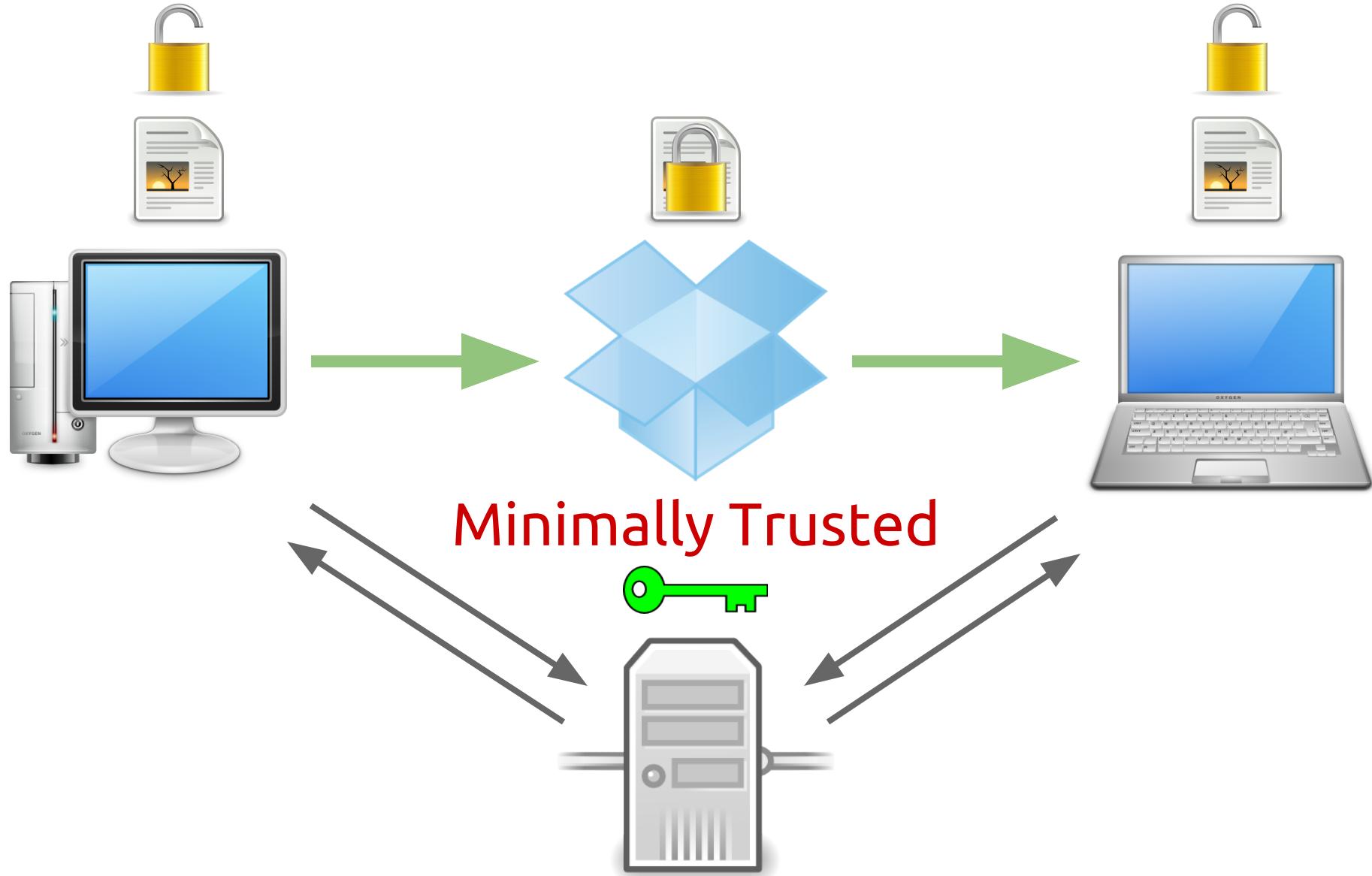
Multi-Device File Sync + Encryption + SaaS



Multi-Device File Sync + Encryption + SaaS



Multi-Device File Sync + Encryption + SaaS



SSaaS Economics

Commoditize Trust

Commoditize Trust

Market of Competing SSPs

Commoditize Trust

Market of Competing SSPs

Reputation

Commoditize Trust

Market of Competing SSPs

Reputation
Security

Commoditize Trust

Market of Competing SSPs

Reputation
Security
Insurance

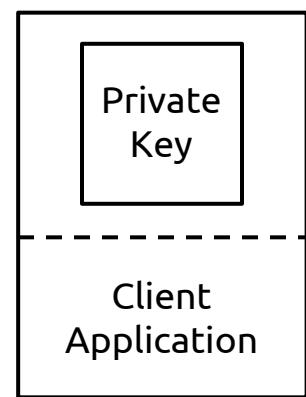
Commoditize Trust

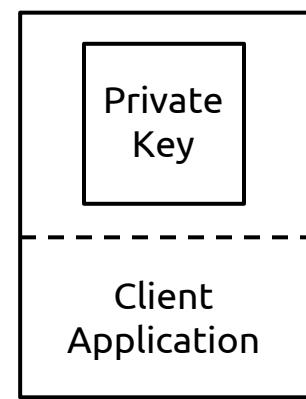
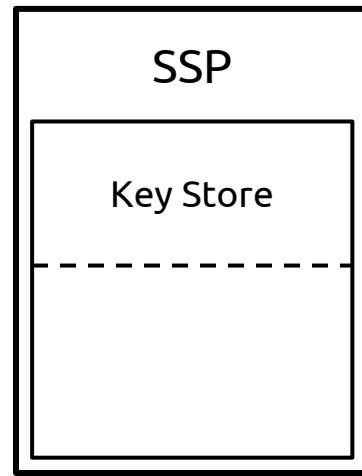
Market of Competing SSPs

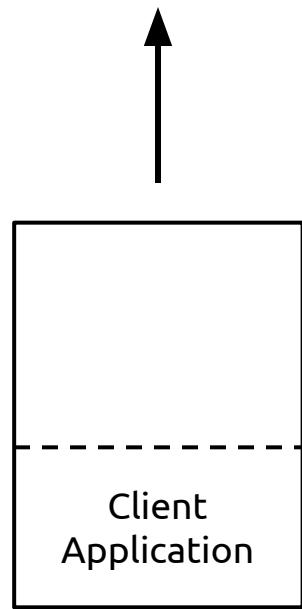
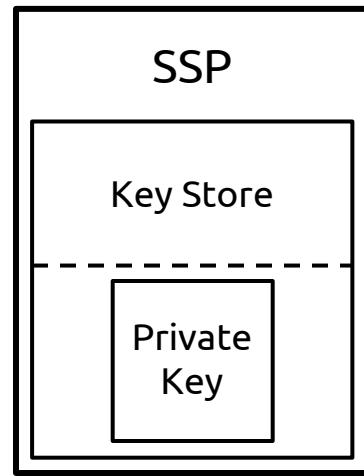
Reputation
Security
Insurance
Price

SSaaS

Security & Trust







Should we trust
a single provider?

Maybe.

Maybe.

Incentives aligned with upholding trust

Maybe.

Incentives aligned with upholding trust

Reputation at stake

Maybe.

Incentives aligned with upholding trust

Reputation at stake

Still a “minimally trusted” entity

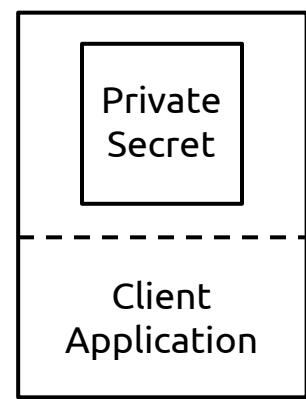
Must we trust
a single provider?

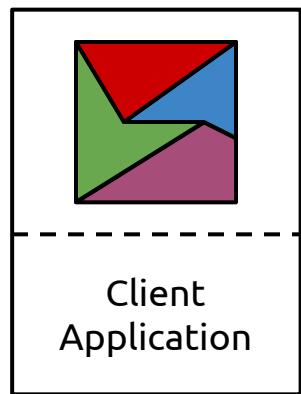
No.

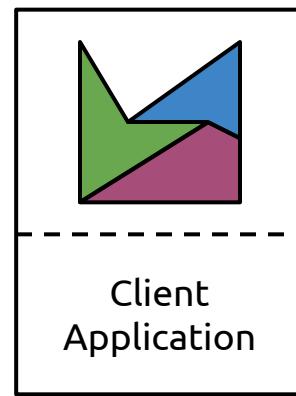
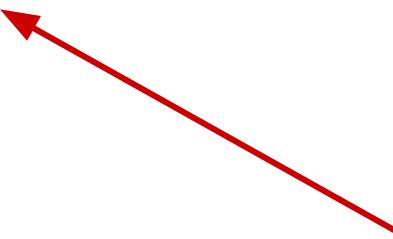
Multi-Provider Sharding

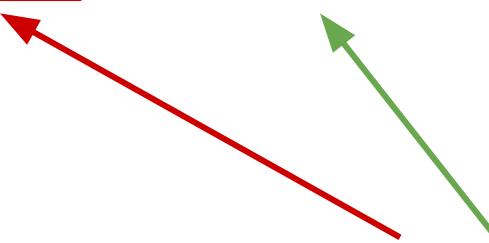
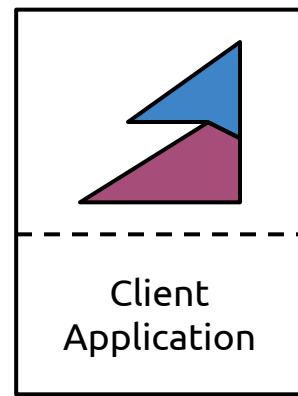
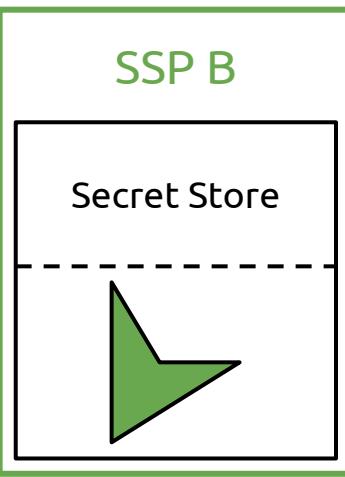
Multi-Provider Sharding

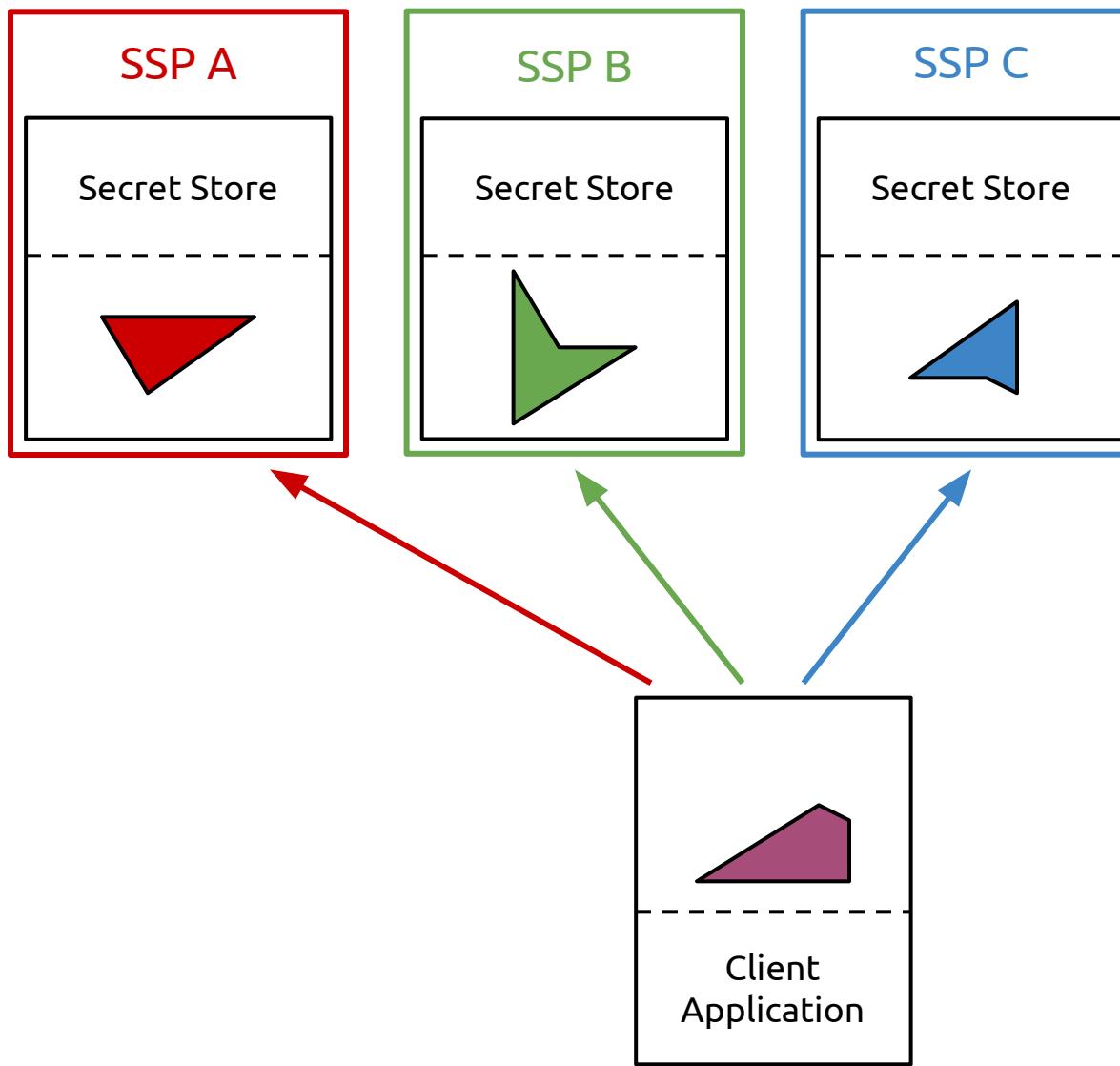
Shamir Secret Sharing

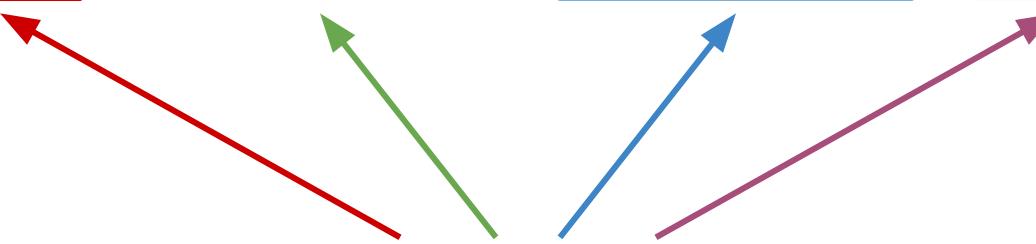
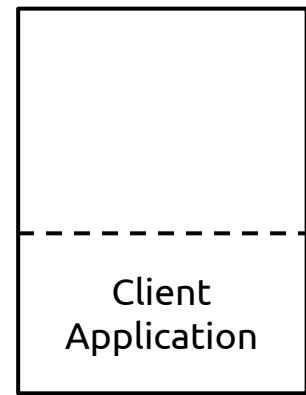
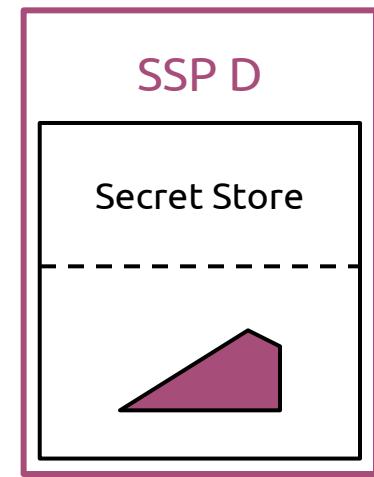
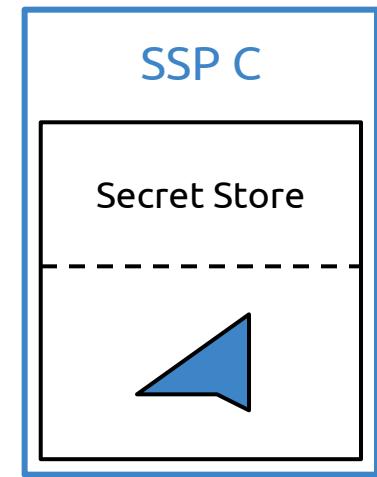
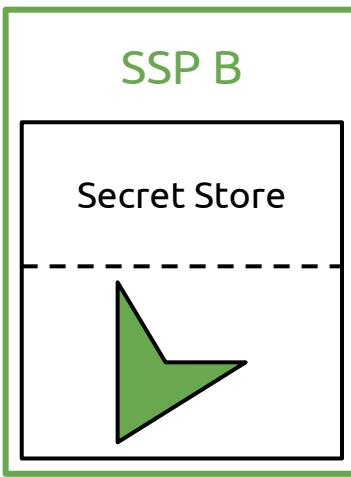


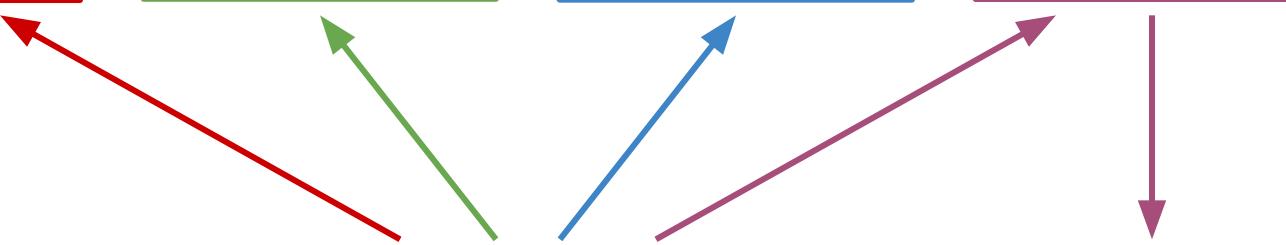
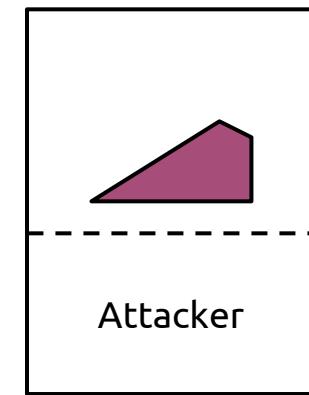
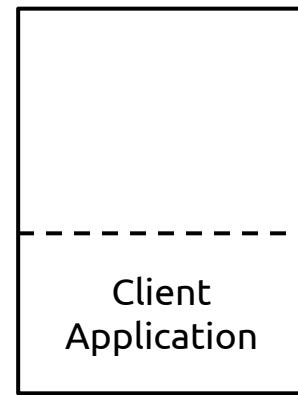
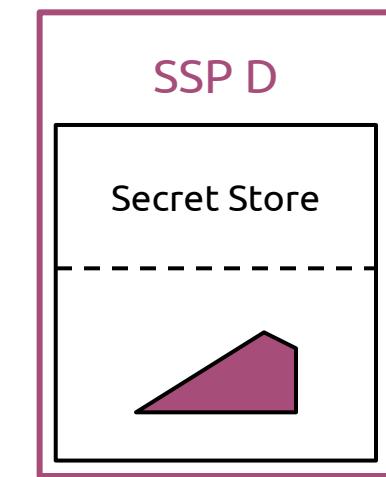
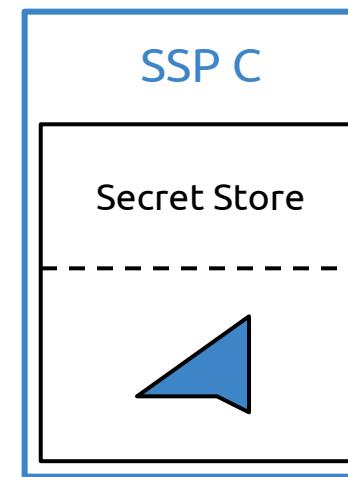
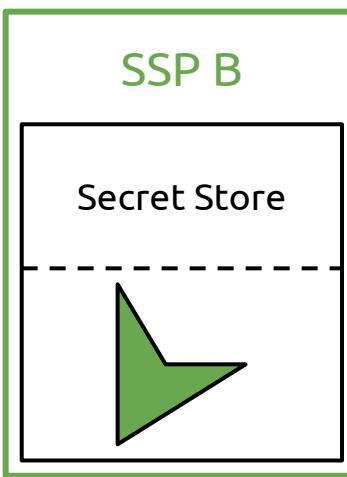


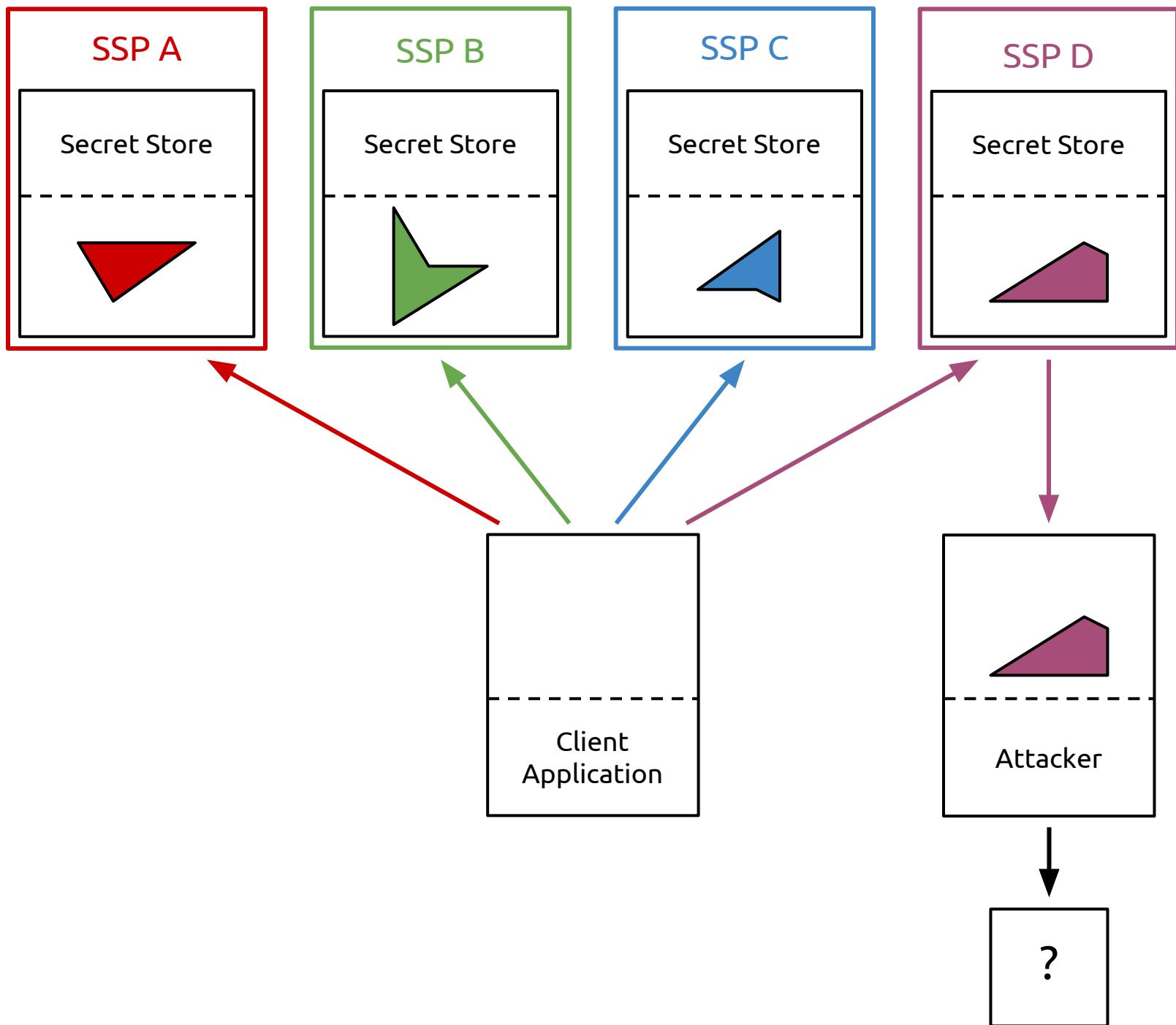












No Single Trusted Third Party

SSP Selection Criteria

SSP Selection Criteria

- + Reputation
- + Security
- + Insurance
- + Cost

SSP Selection Criteria

- + Reputation
- + Security
- + Insurance
- + Cost
- + Geopolitical Diversity

SSP Selection Criteria

- + Reputation
- + Security
- + Insurance
- + Cost
- + Geopolitical Diversity
- + Ownership Diversity

Custos Prototype

Custos Prototype

Latin for “Guard”

“Key Storage as a Service” (KSaaS)

Custos SSP Server

Custos SSP Server

Client A

Client B

Client C

Custos SSP Server

Secret Store

Client A

Client B

Client C

Custos SSP Server

Secret Store

Application

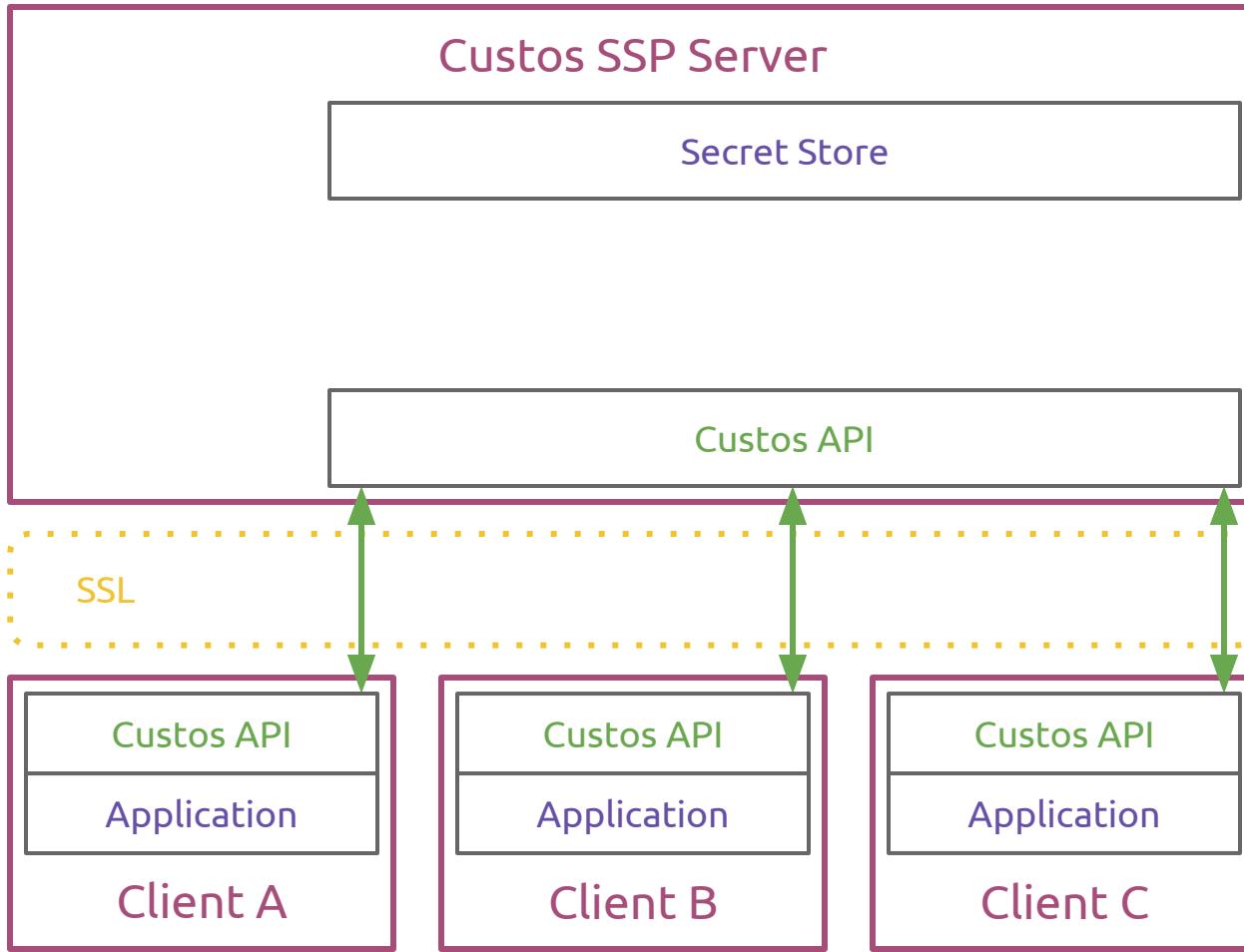
Client A

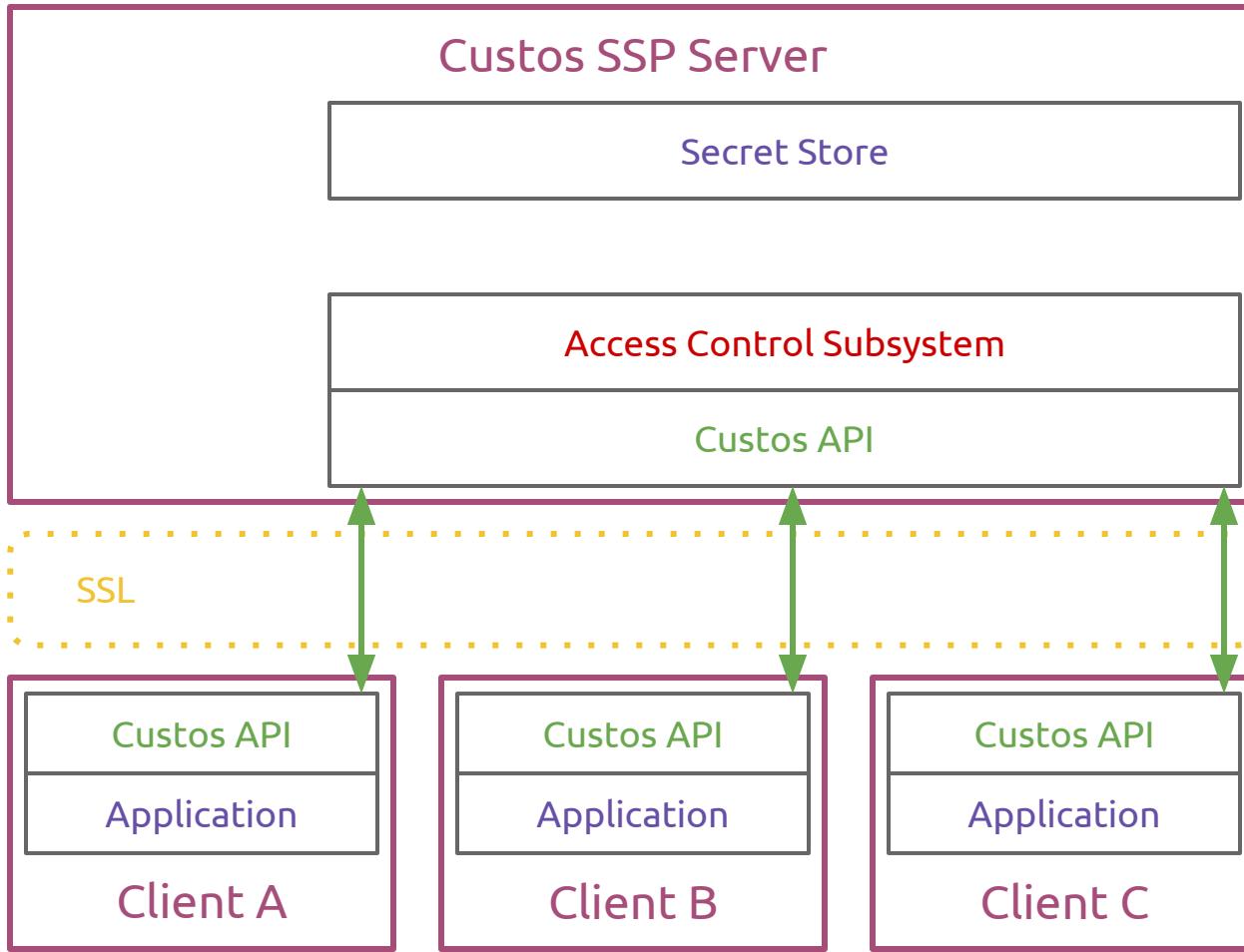
Application

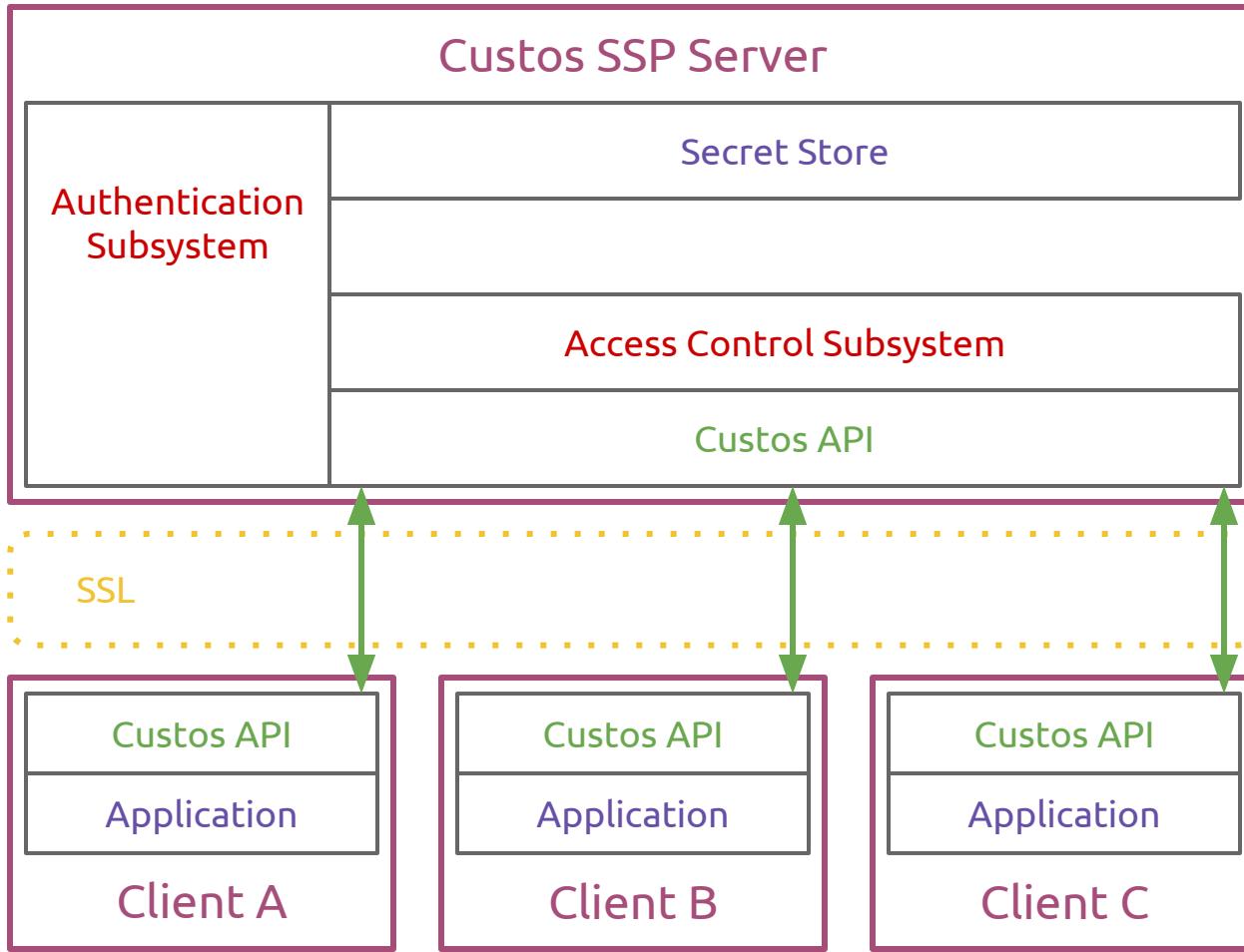
Client B

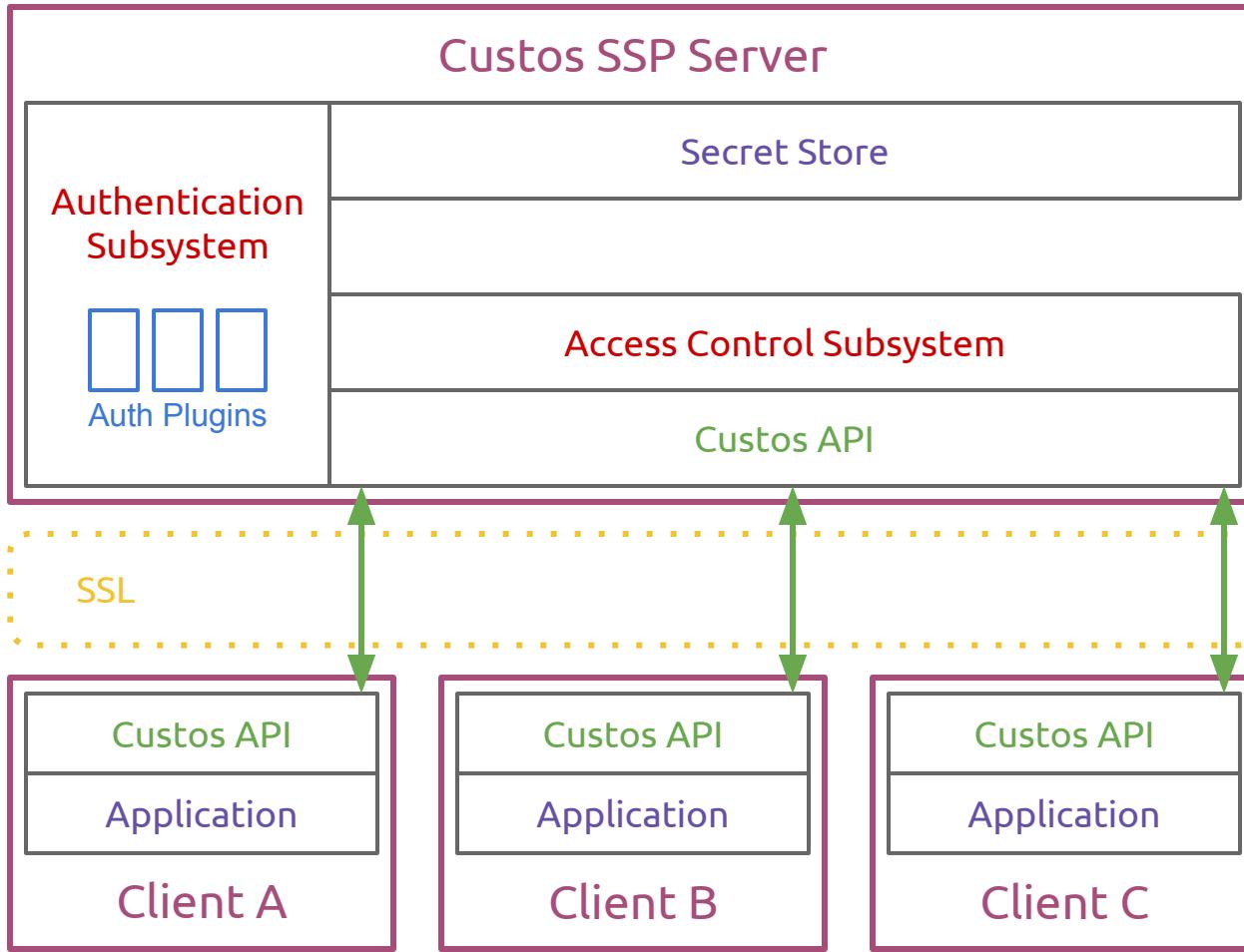
Application

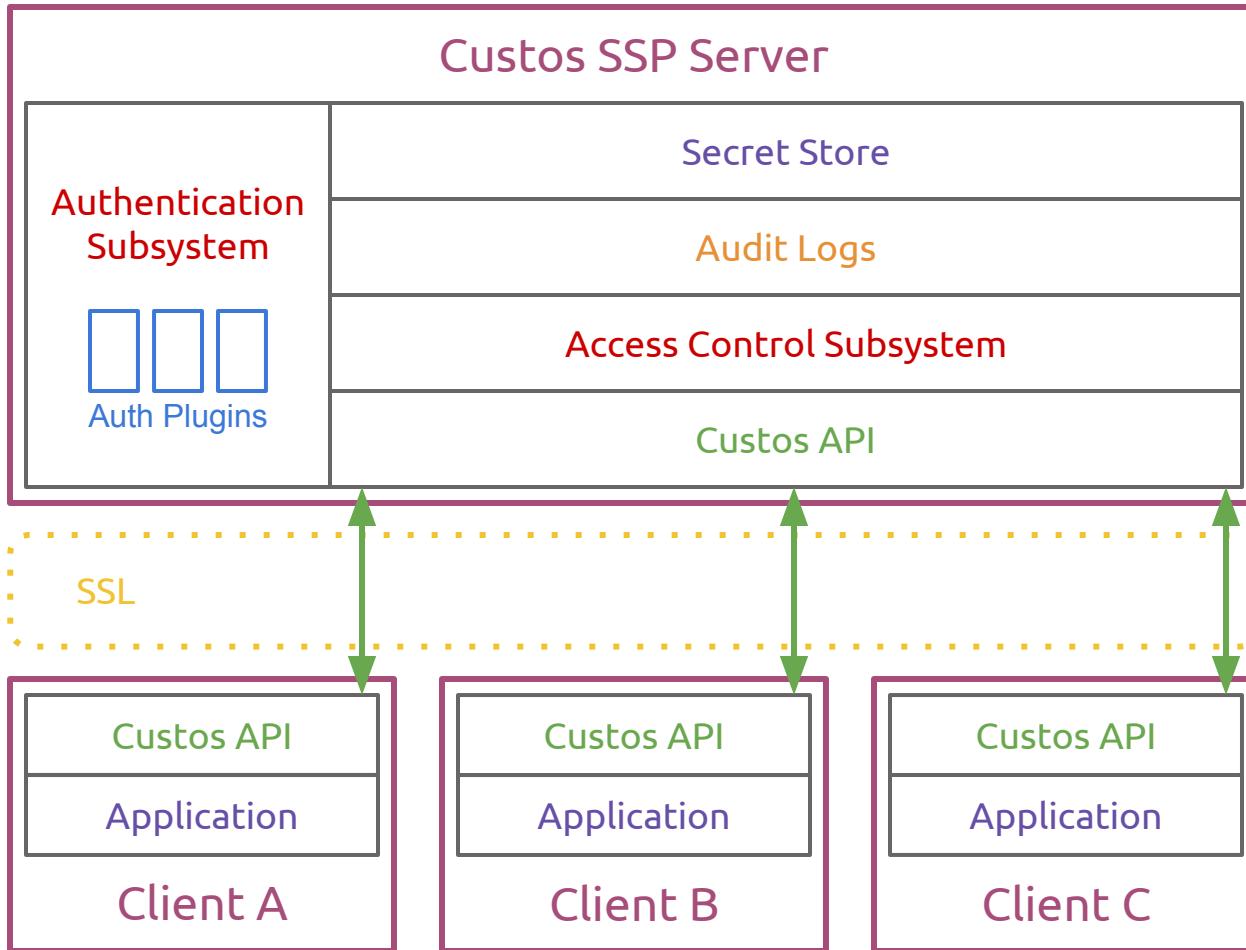
Client C



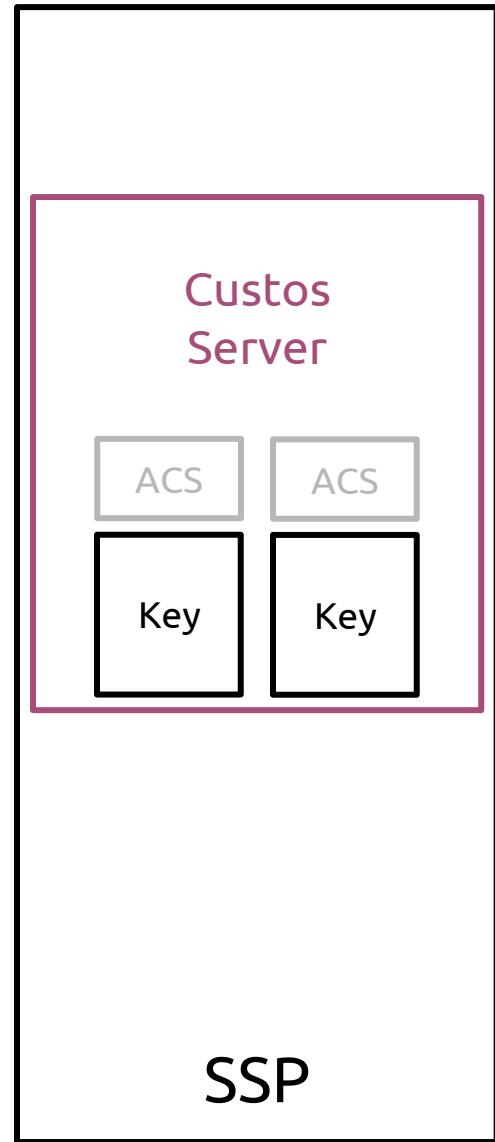


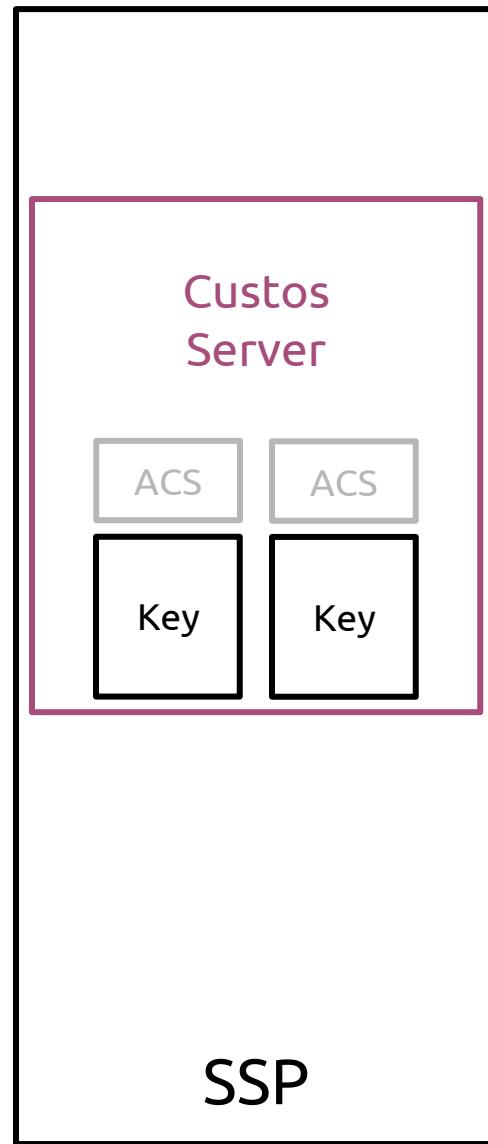
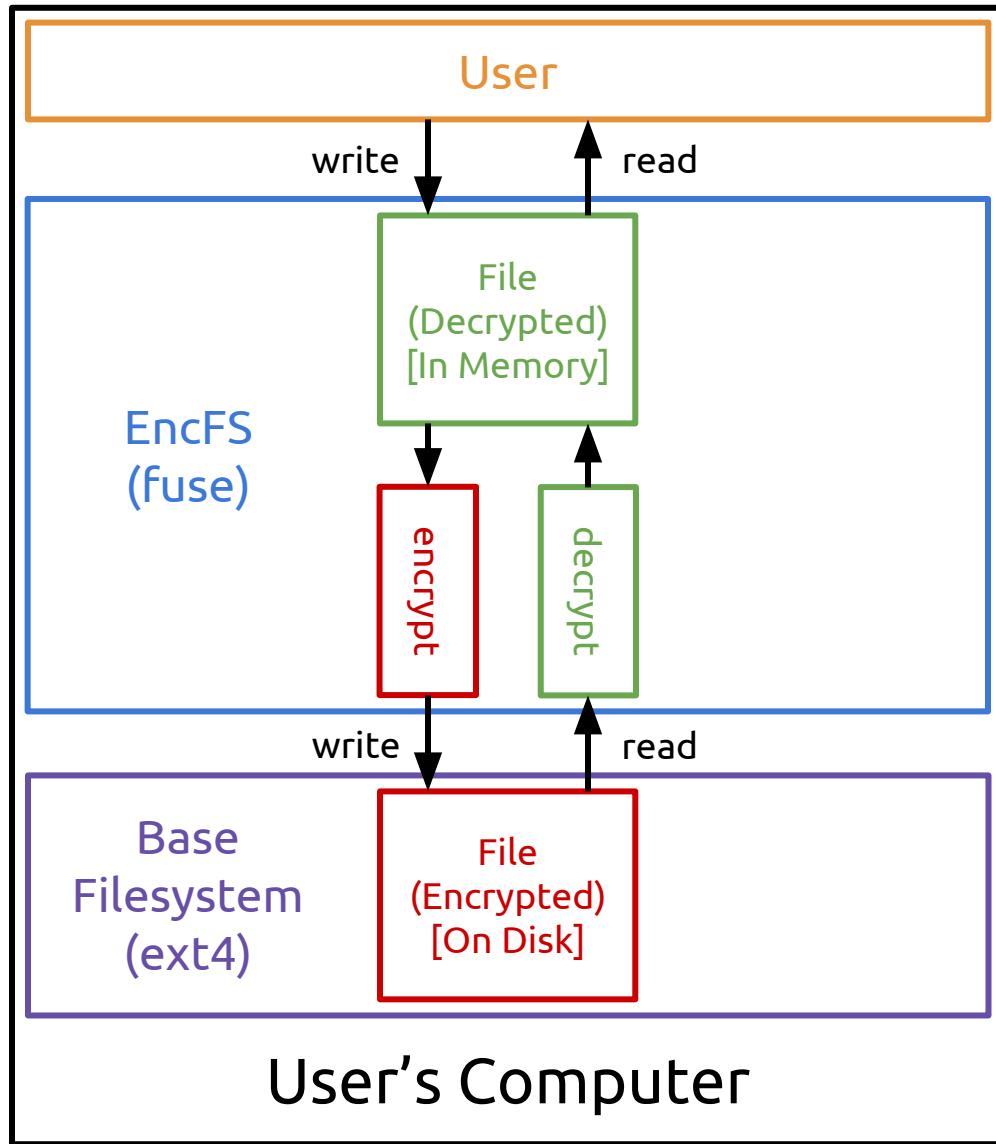


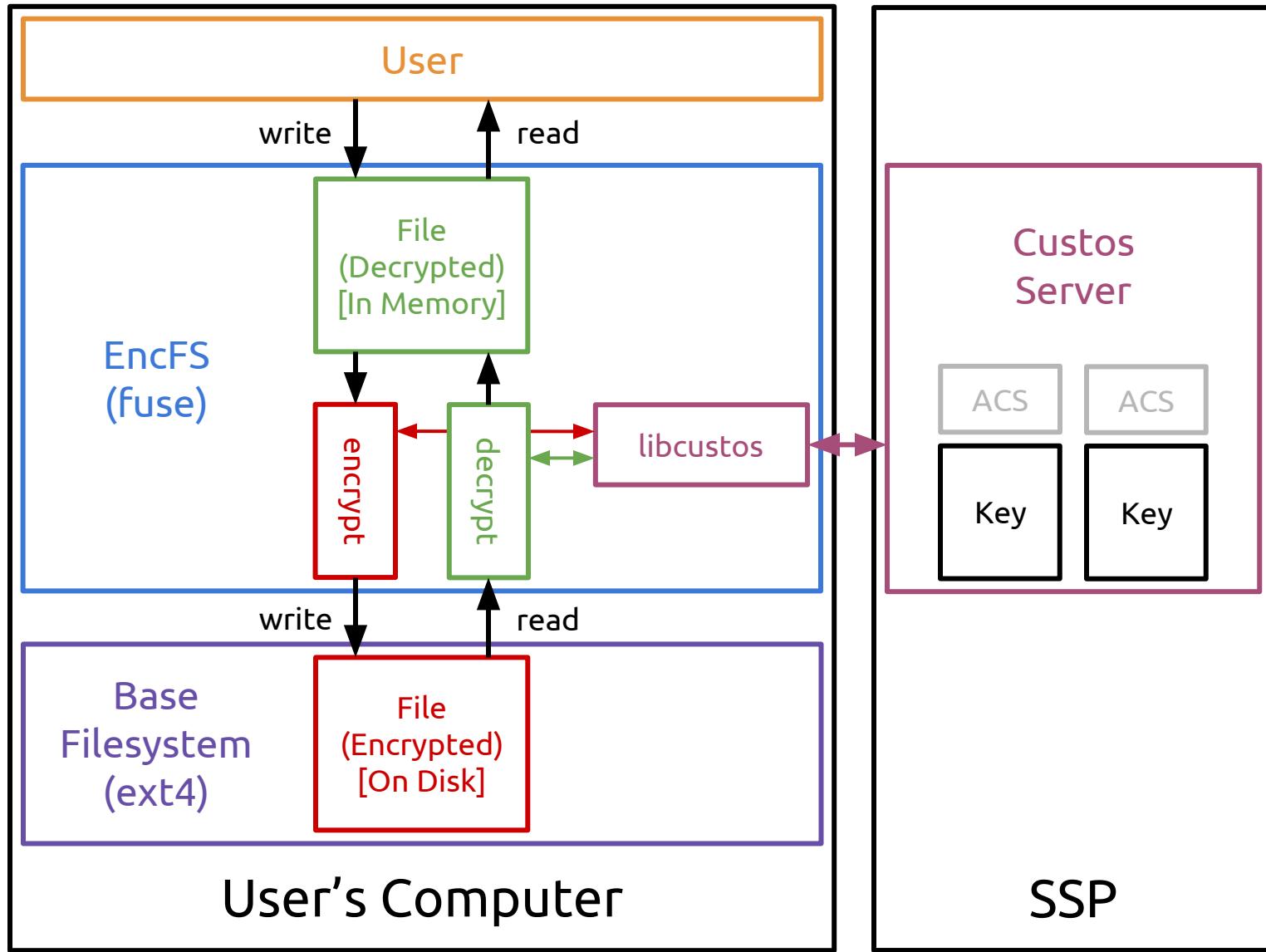


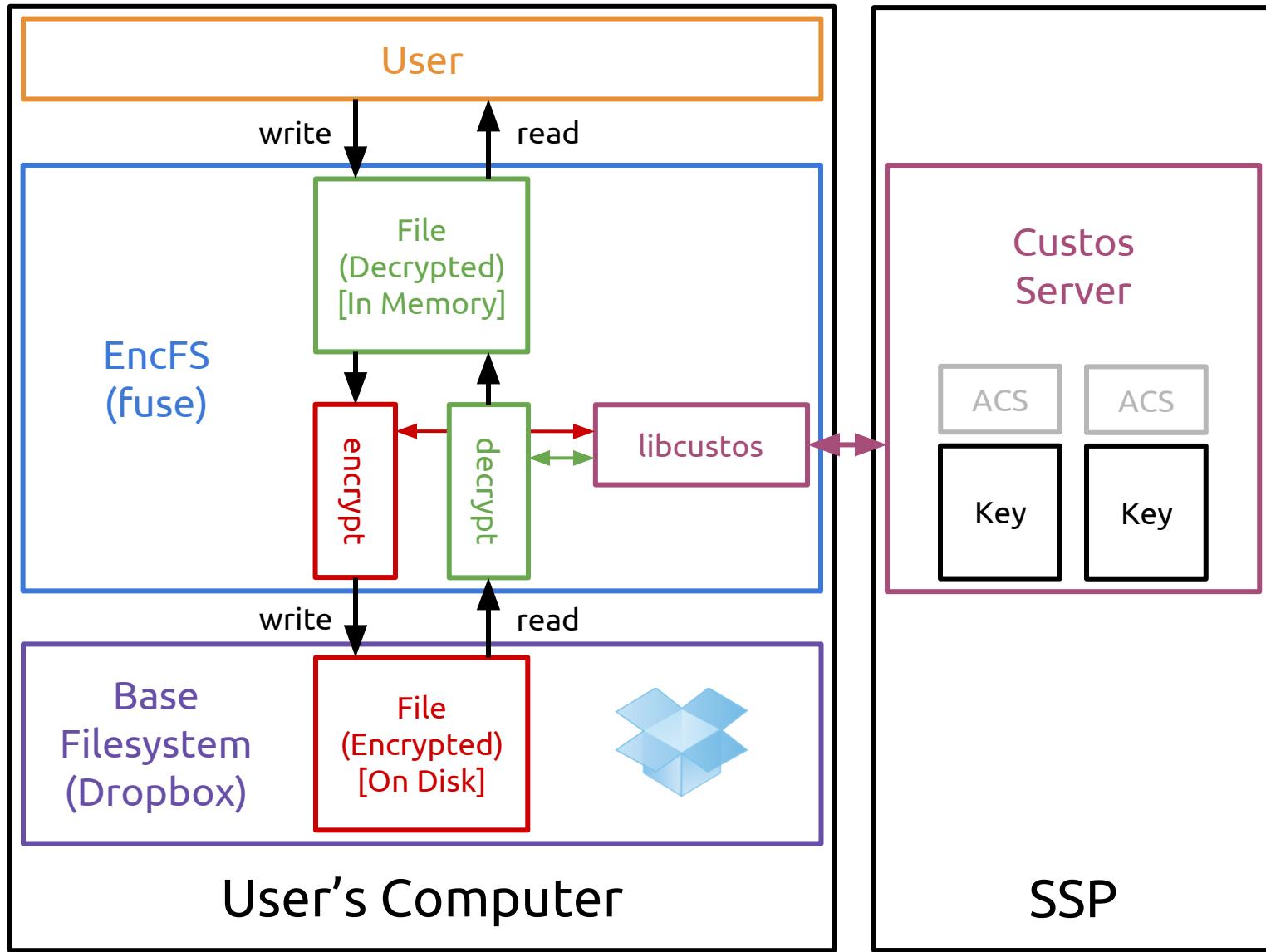


EncFS: Custos-Backed Encrypted File System







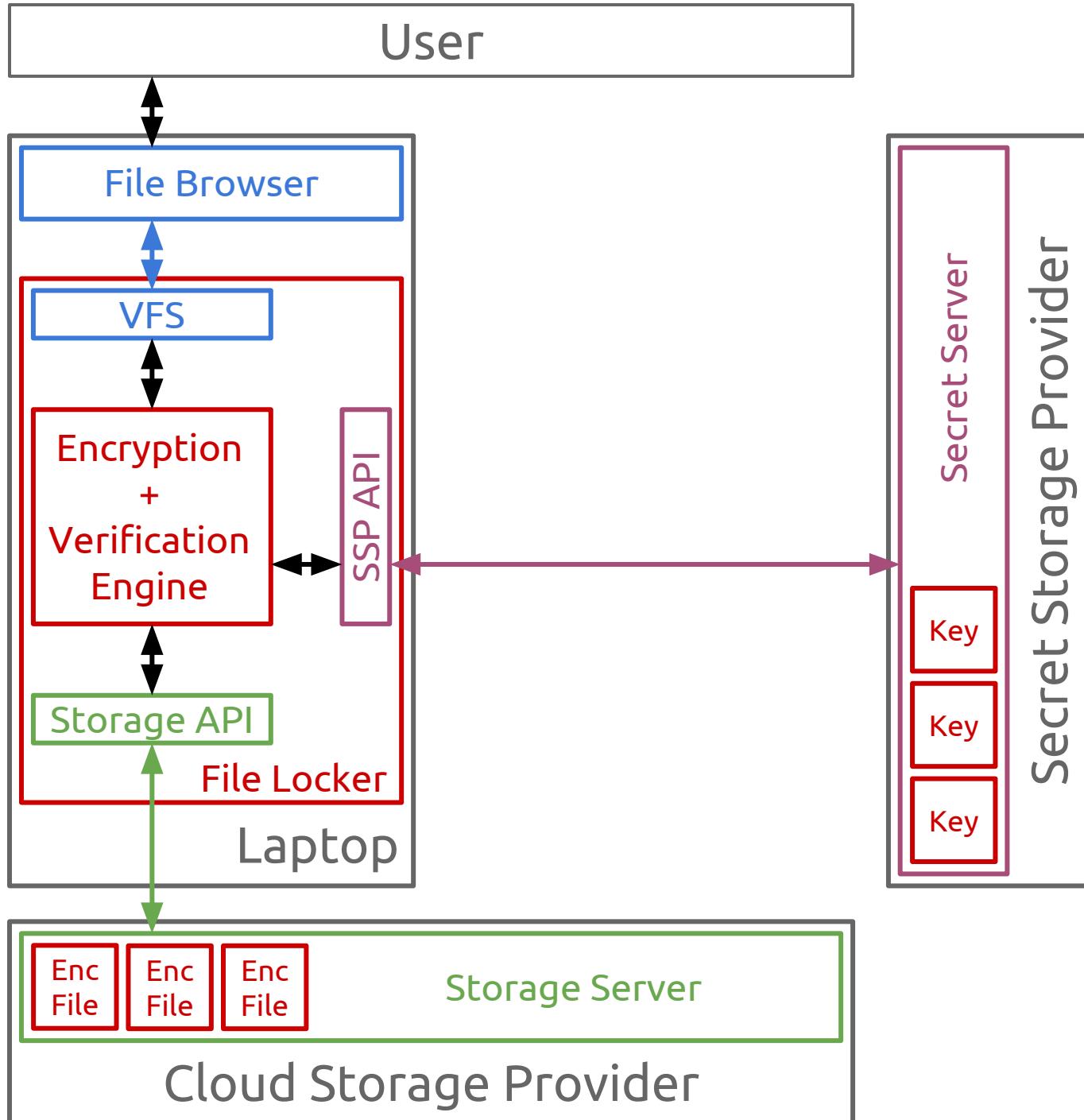


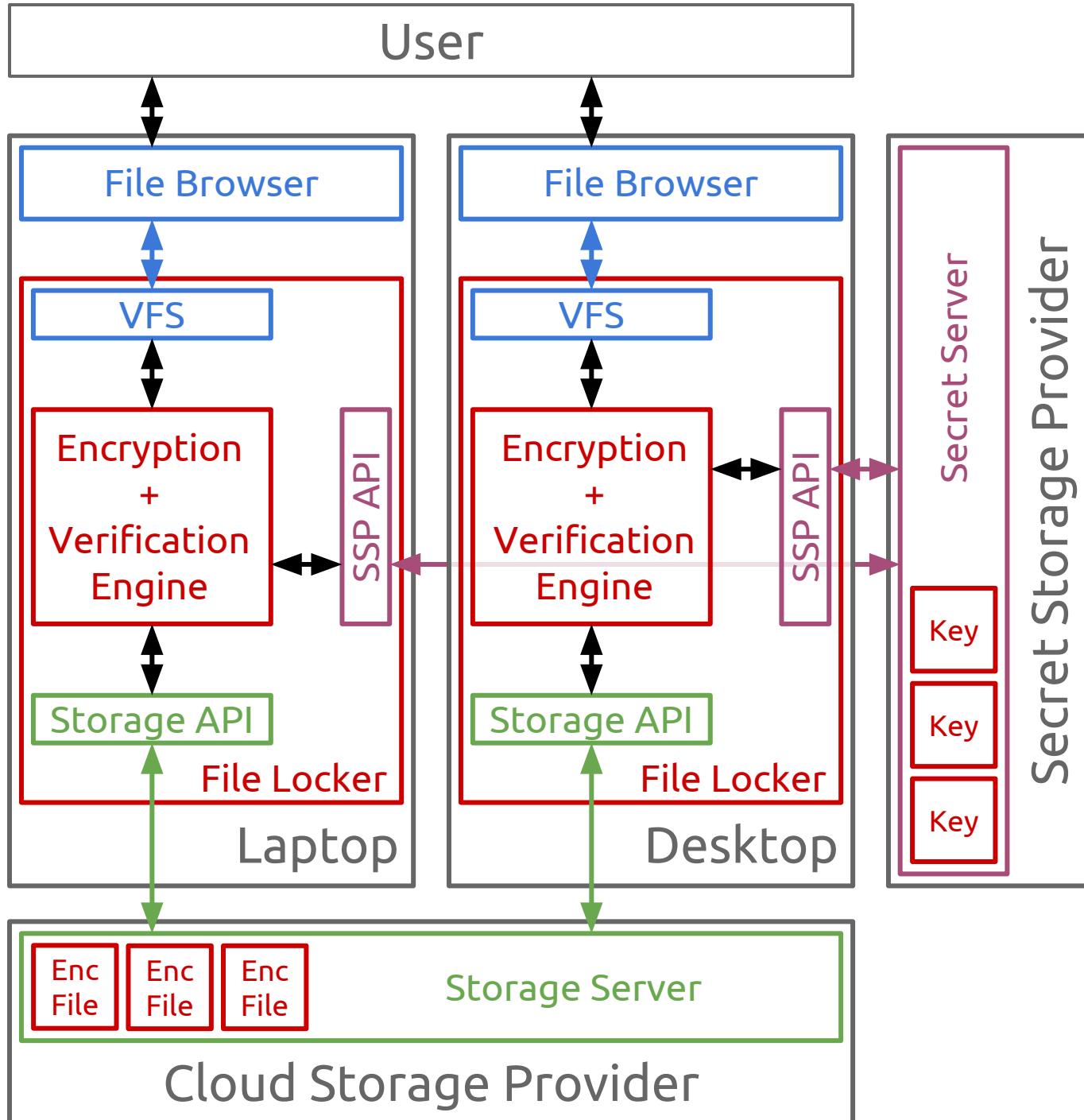
SSaaS Applications

Storage Applications
Communication Applications
Authentication Applications
Crypto Processing Applications

Storage Applications

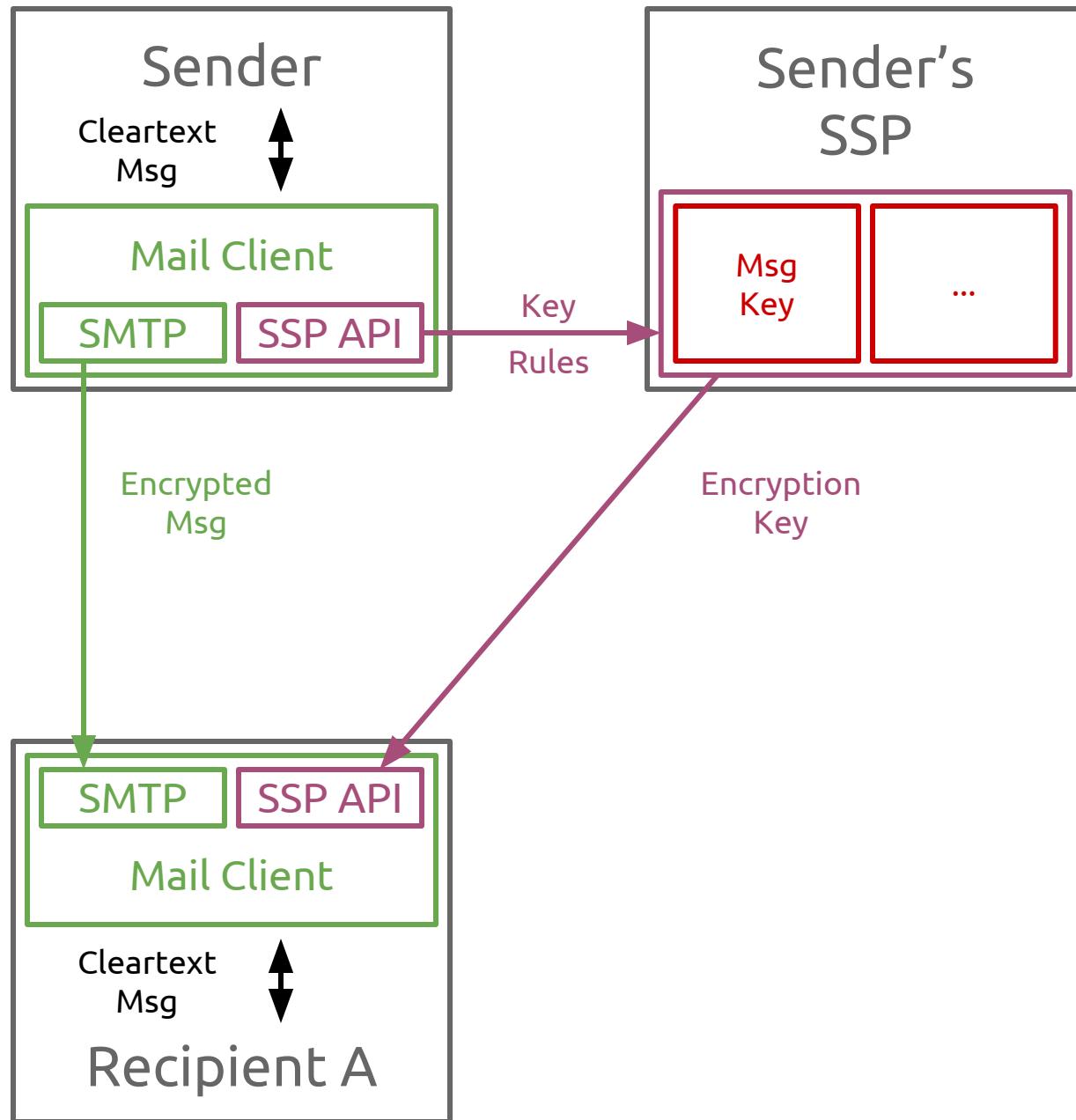
Client-Encrypted File Locker

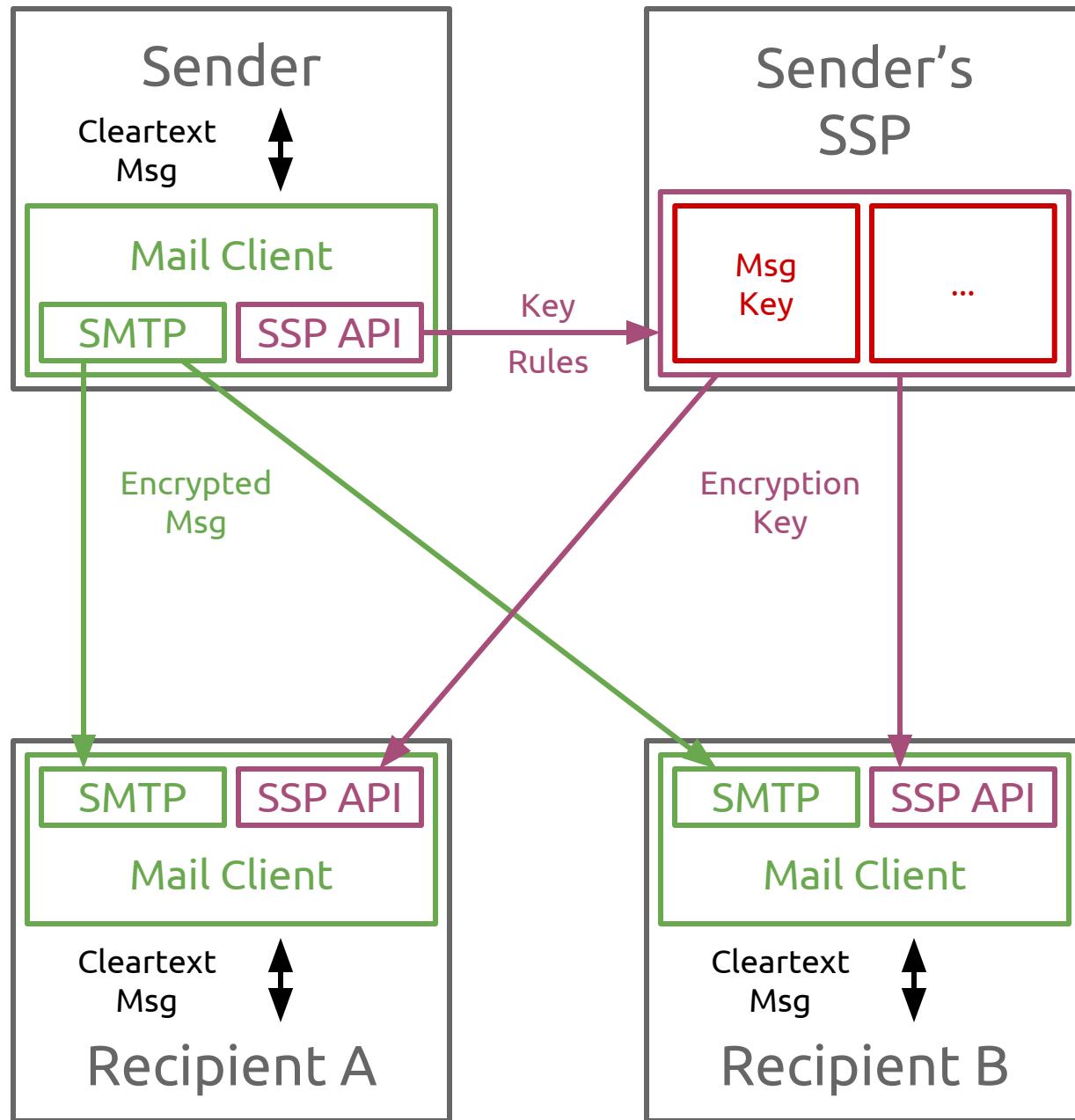




Communication Applications

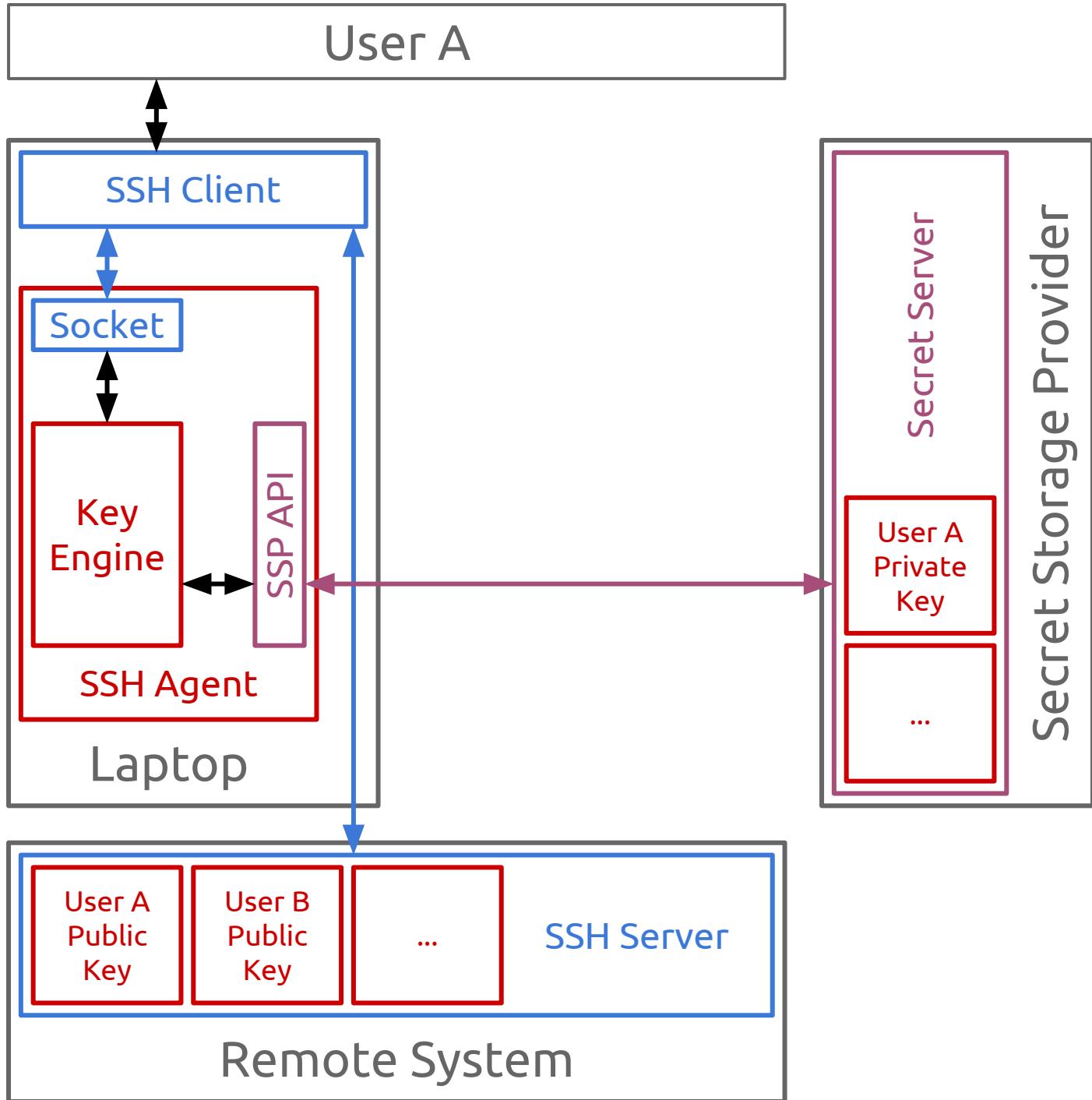
Encrypted Email

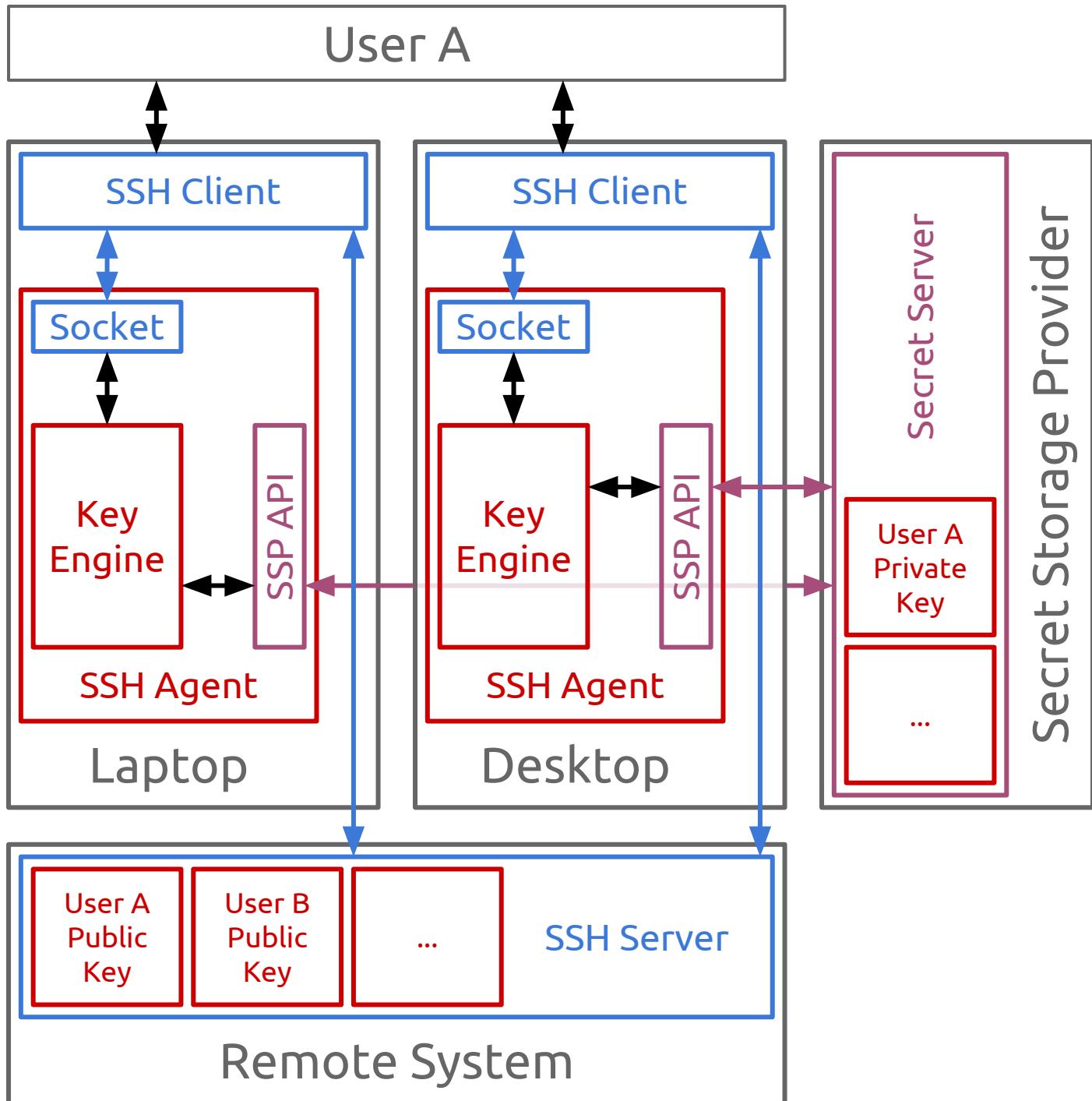




Authentication Applications

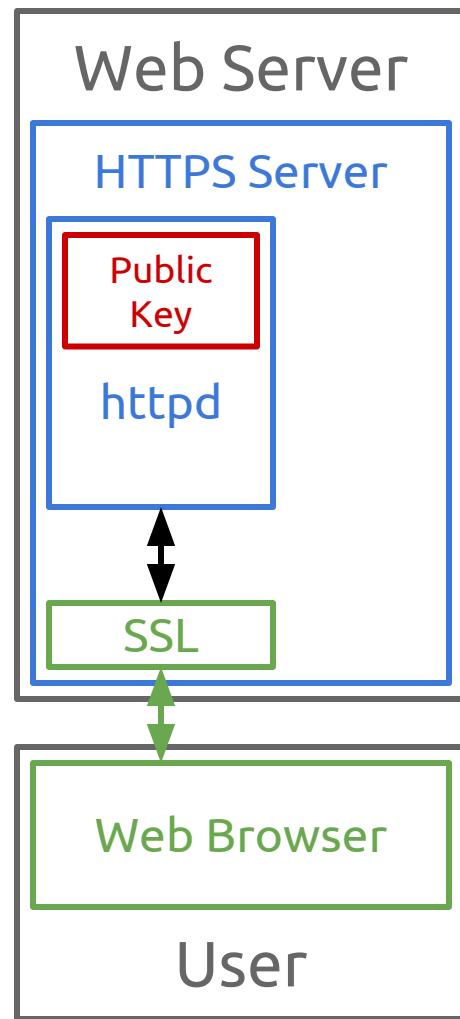
Multi-Device & Managed SSH Agent

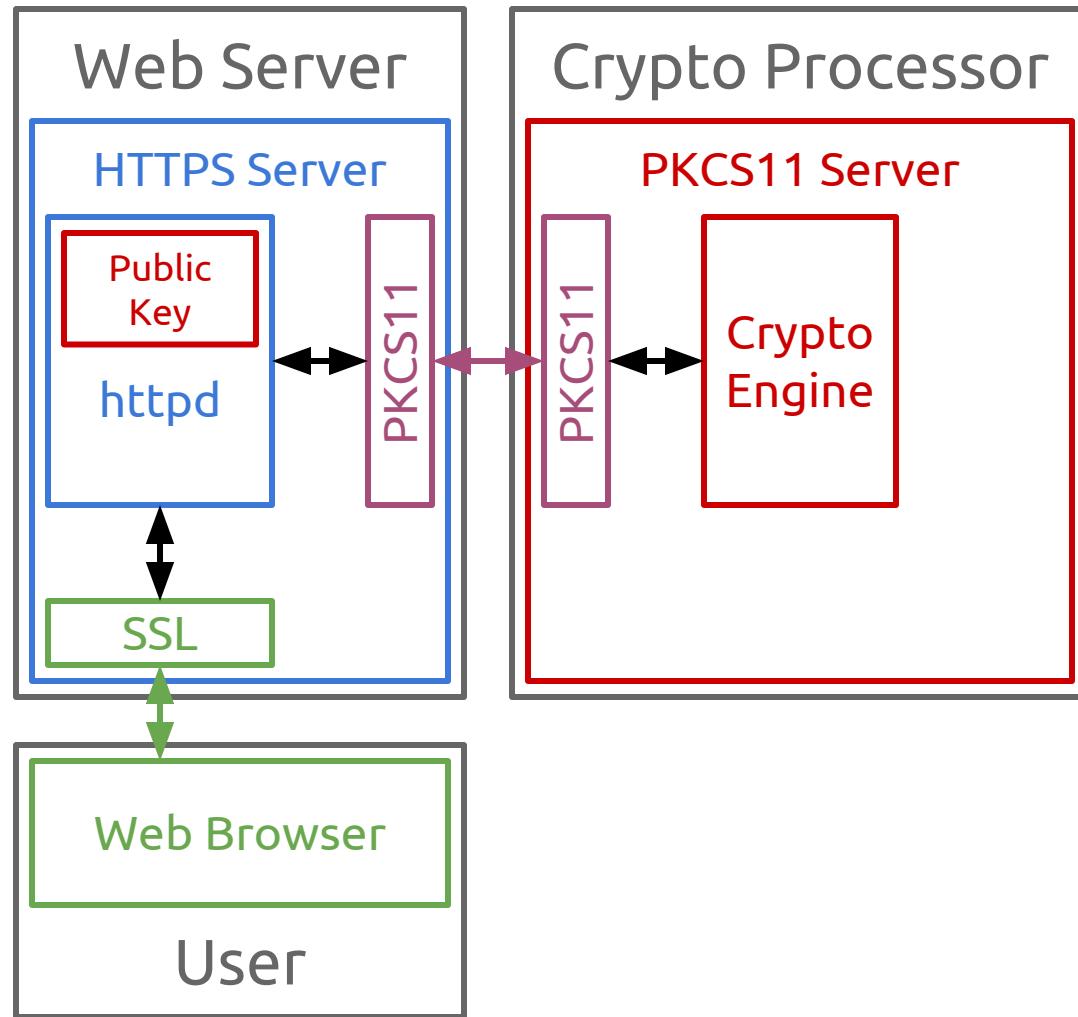


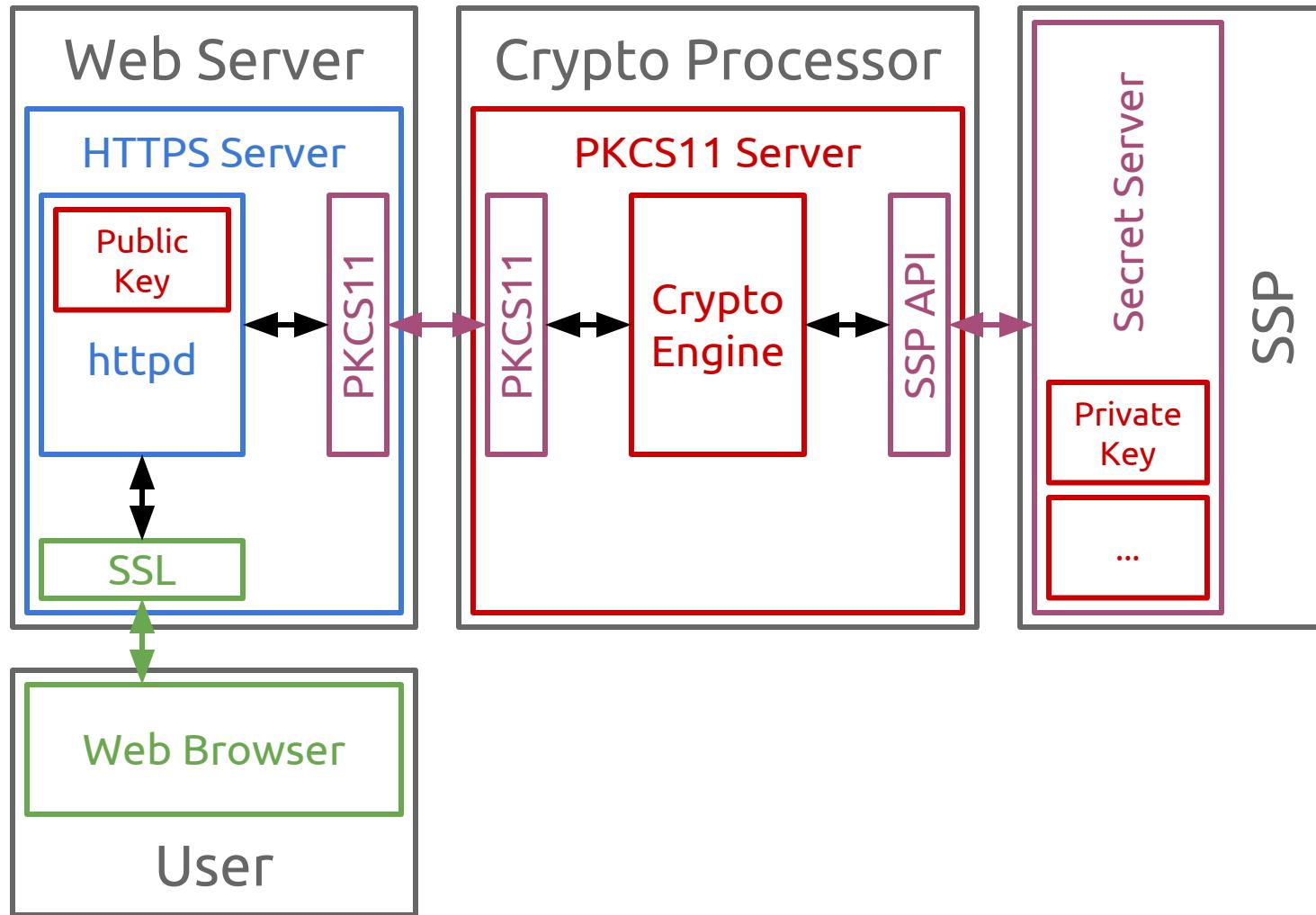


Crypto Processing Applications

Dedicated Crypto Processor







Proposed Work

1. Trust Analysis Surveys
2. Implementation Extensions
3. SSaaS Analysis

1. Trust Analysis Surveys

Consumer Services

Consumer Services

Dropbox BitTorrent Sync

SpiderOak Mint

OnePassword

LastPass

Consumer Services

Dropbox BitTorrent Sync

SpiderOak Mint

OnePassword

LastPass

Developer Services

Consumer Services

Dropbox BitTorrent Sync

SpiderOak Mint

OnePassword

LastPass

Developer Services

Amazon EC2

Google Compute Engine

Javascript Crypto

Puppet

Chef

KeyBase.io

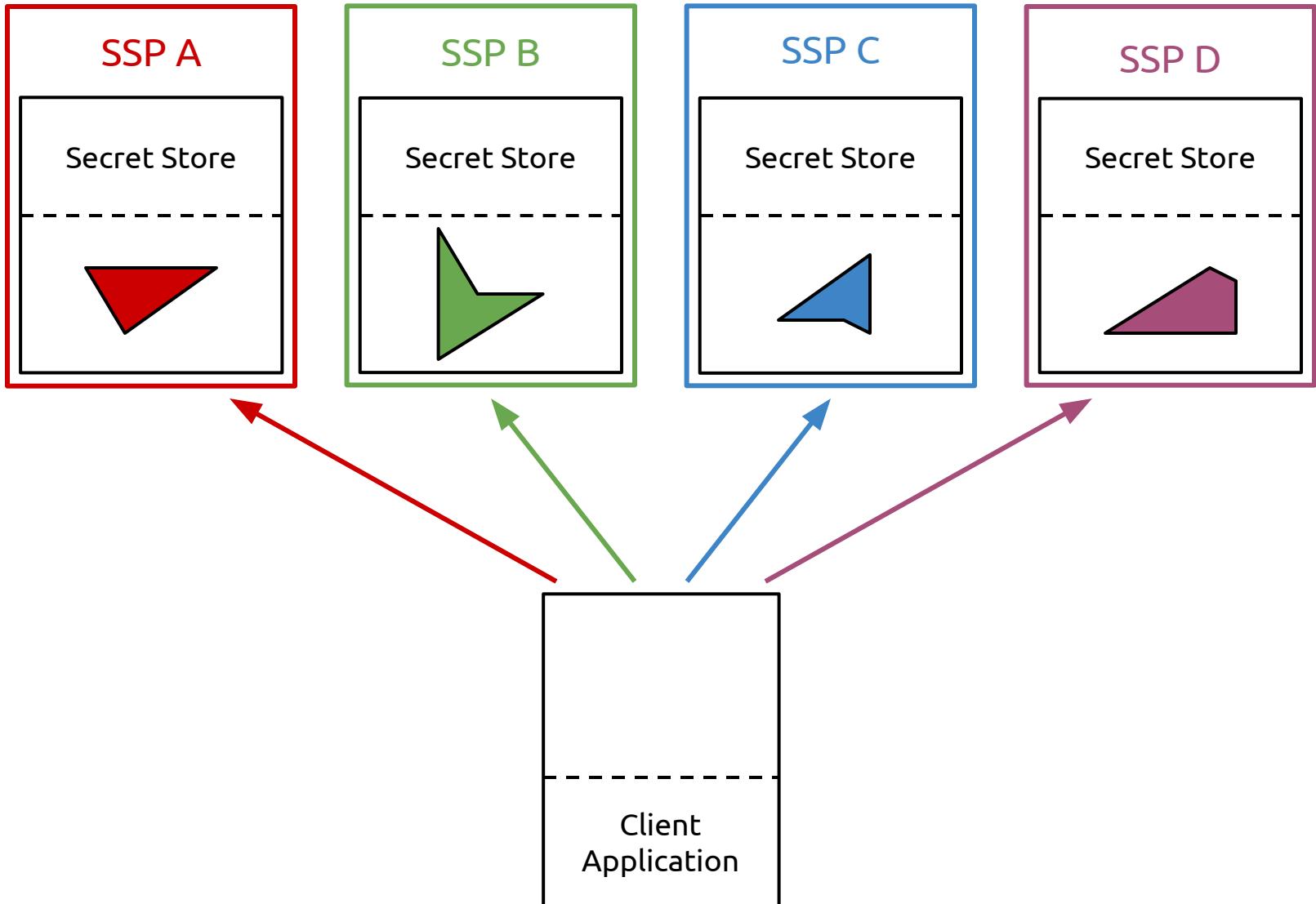
OpenStack Barbican

Amazon CloudHSM

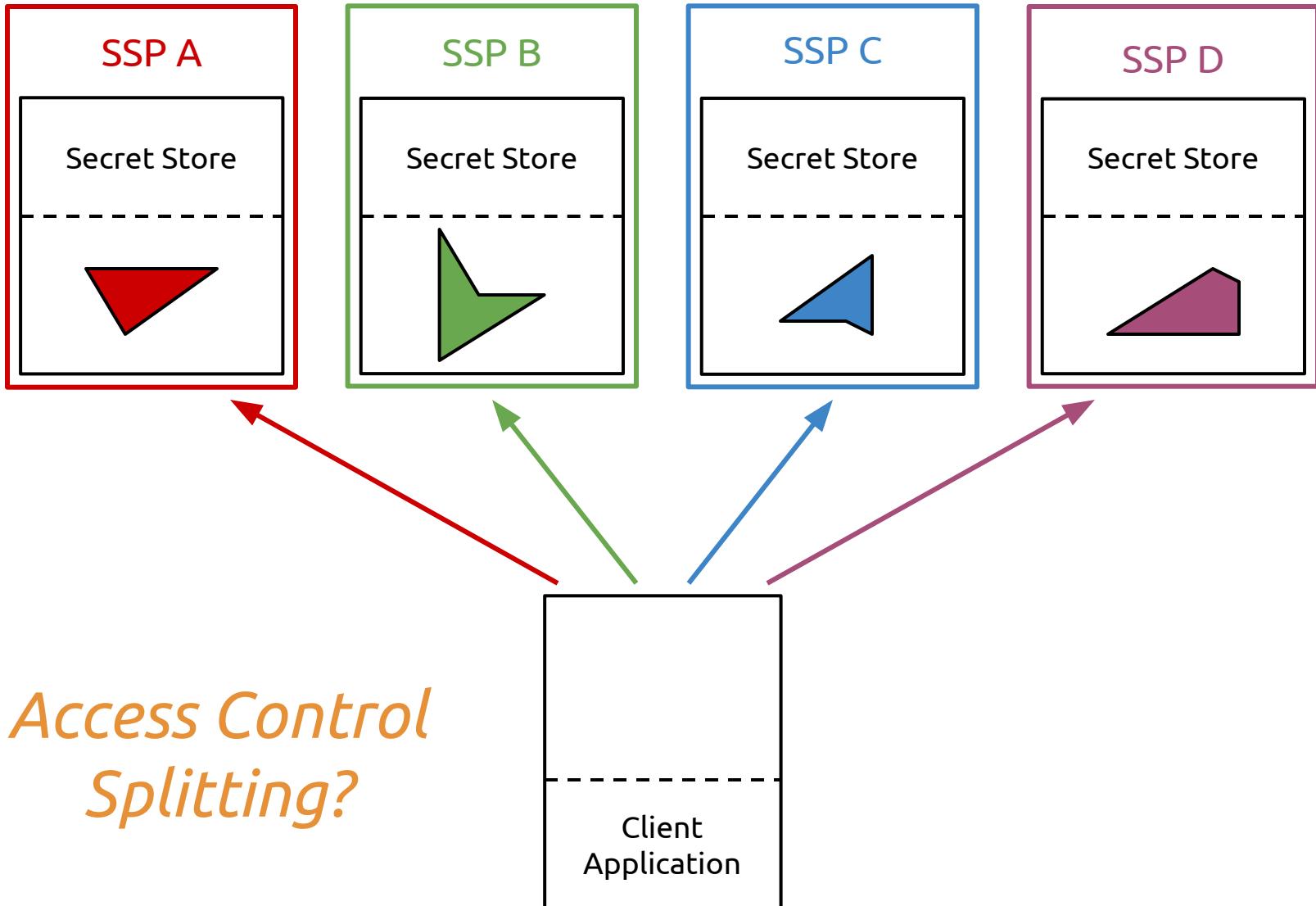
2. Implementation Extensions

SSP Improvements

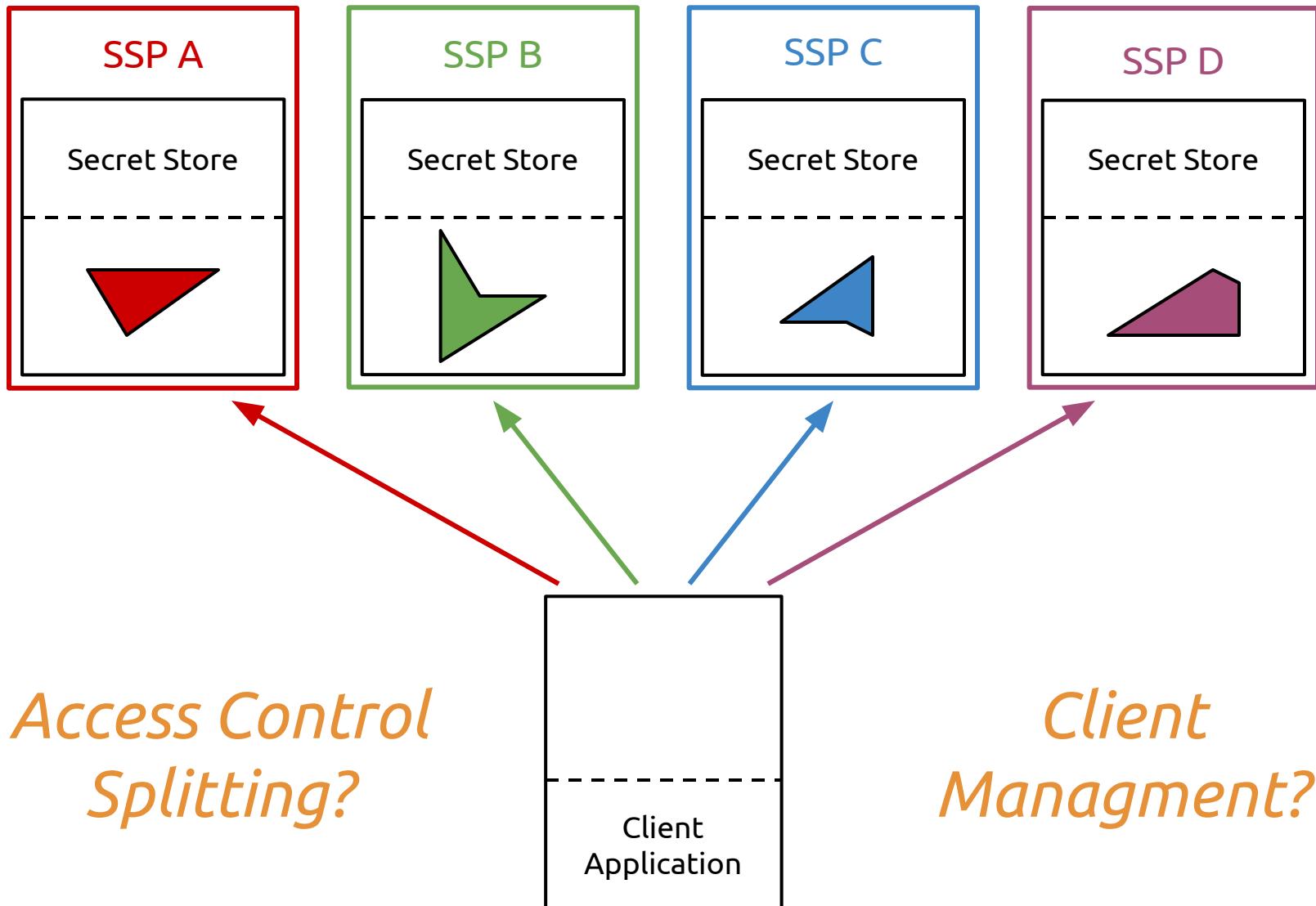
SSP Improvements



SSP Improvements



SSP Improvements



SSP Improvements

SSP Improvements

Improved Access Control Implementation

SSP Improvements

Improved Access Control Implementation

Higher Performance Storage Backend

SSP Improvements

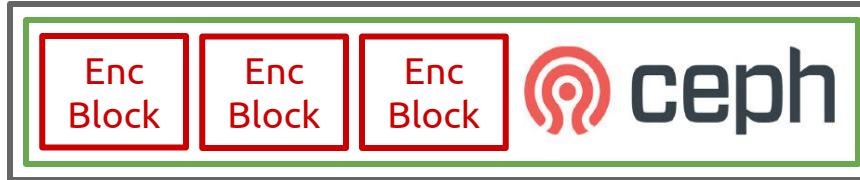
Improved Access Control Implementation

Higher Performance Storage Backend

Updated API

Client Build-Out

Client Build-Out



Client Build-Out

VM

Enc
Block

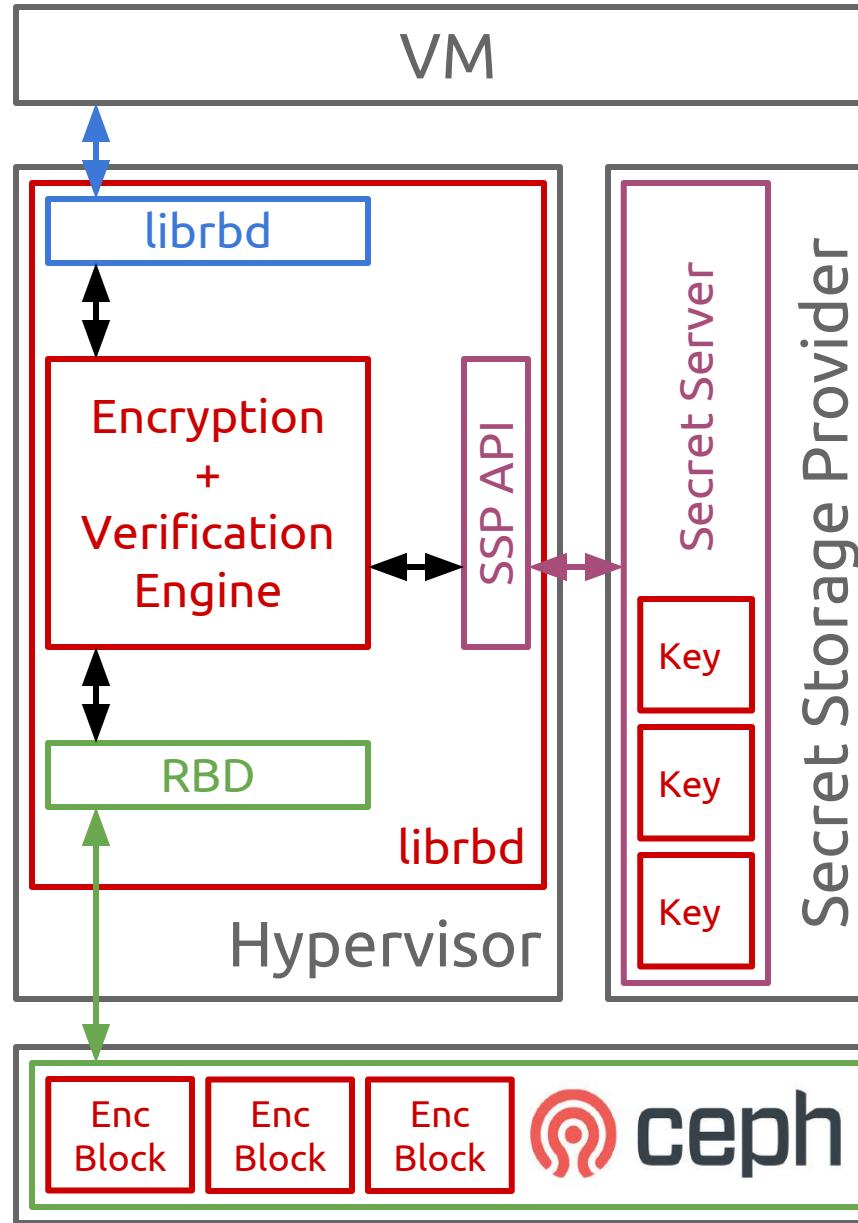
Enc
Block

Enc
Block



ceph

Client Build-Out



Client Build-Out

Client Build-Out

EncFS Improvements

Client Build-Out

EncFS Improvements
(possible) eCryptFS Support

Client Build-Out

EncFS Improvements
(possible) eCryptFS Support
Dropbox Client Support

3. SSaaS Analysis

Capabilities and Features Offered by SSaaS

Capabilities and Features Offered by SSaaS

Tradeoffs and Costs of SSaaS

Capabilities and Features Offered by SSaaS

Tradeoffs and Costs of SSaaS

Trust Modeling for Prototypes

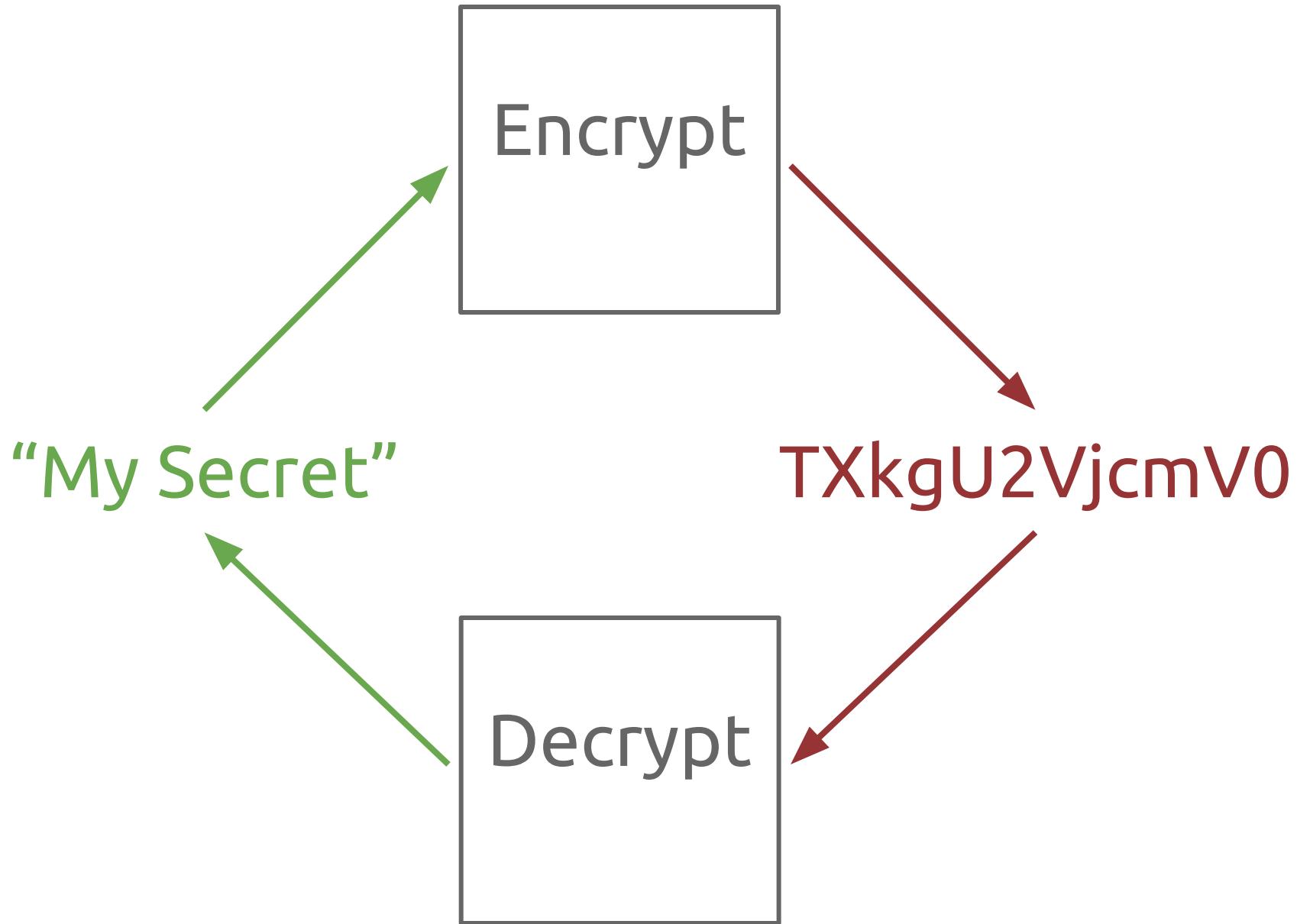
Capabilities and Features Offered by SSaaS
Tradeoffs and Costs of SSaaS
Trust Modeling for Prototypes
Performance of Prototypes

Thank You

Questions?

Extra Slides

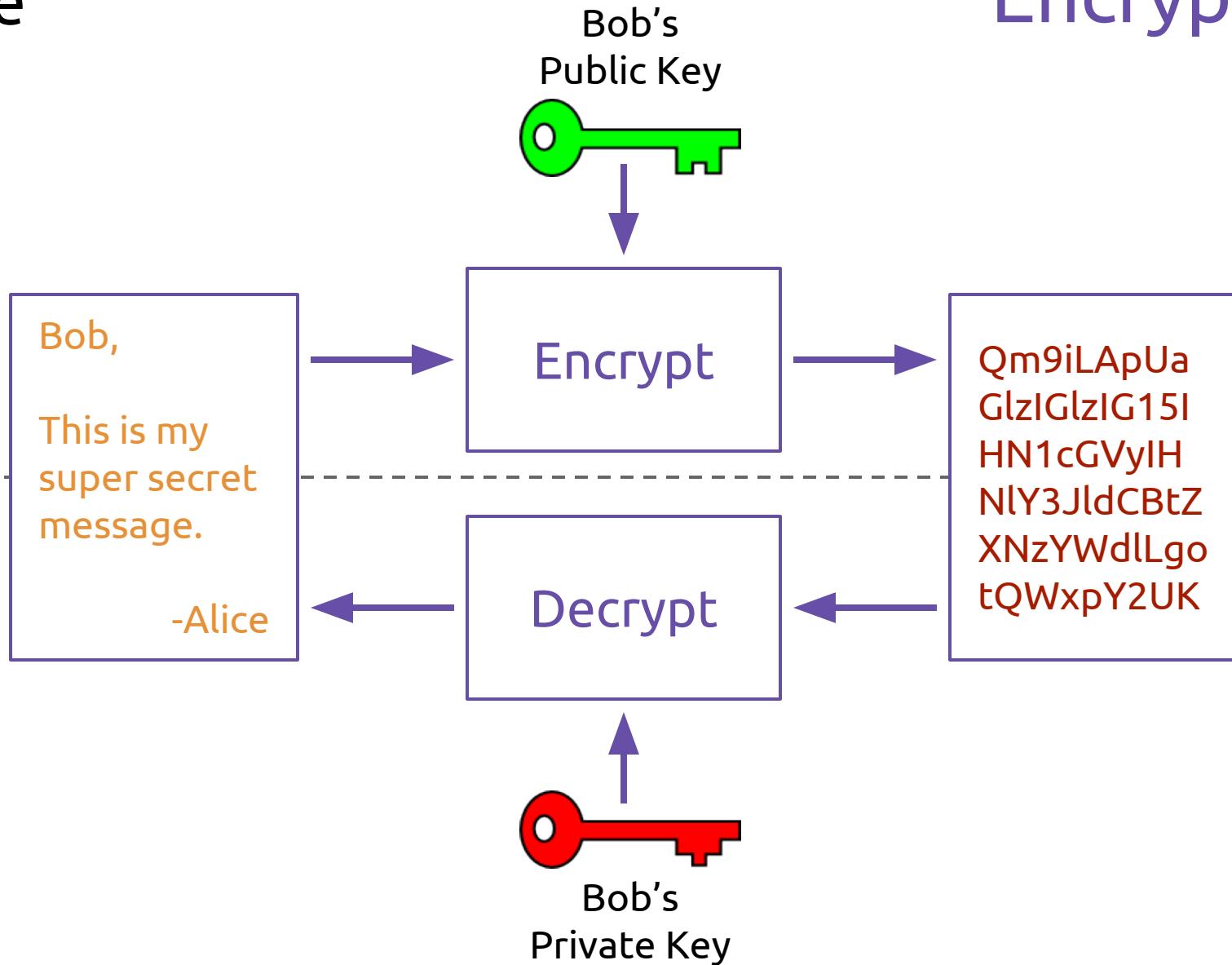
Symmetric Cryptography



Asymmetric Cryptography

Alice

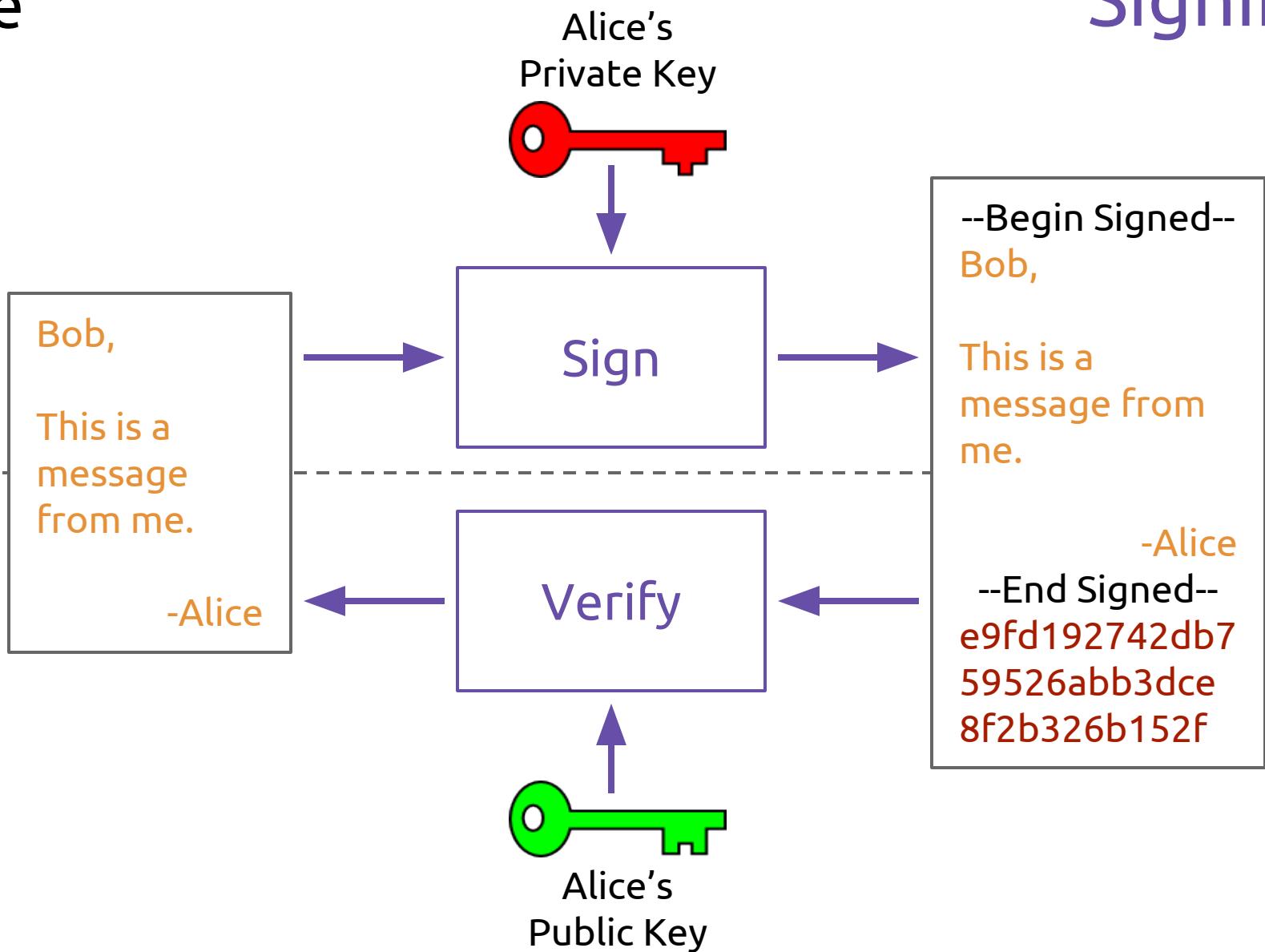
Encryption



Bob

Alice

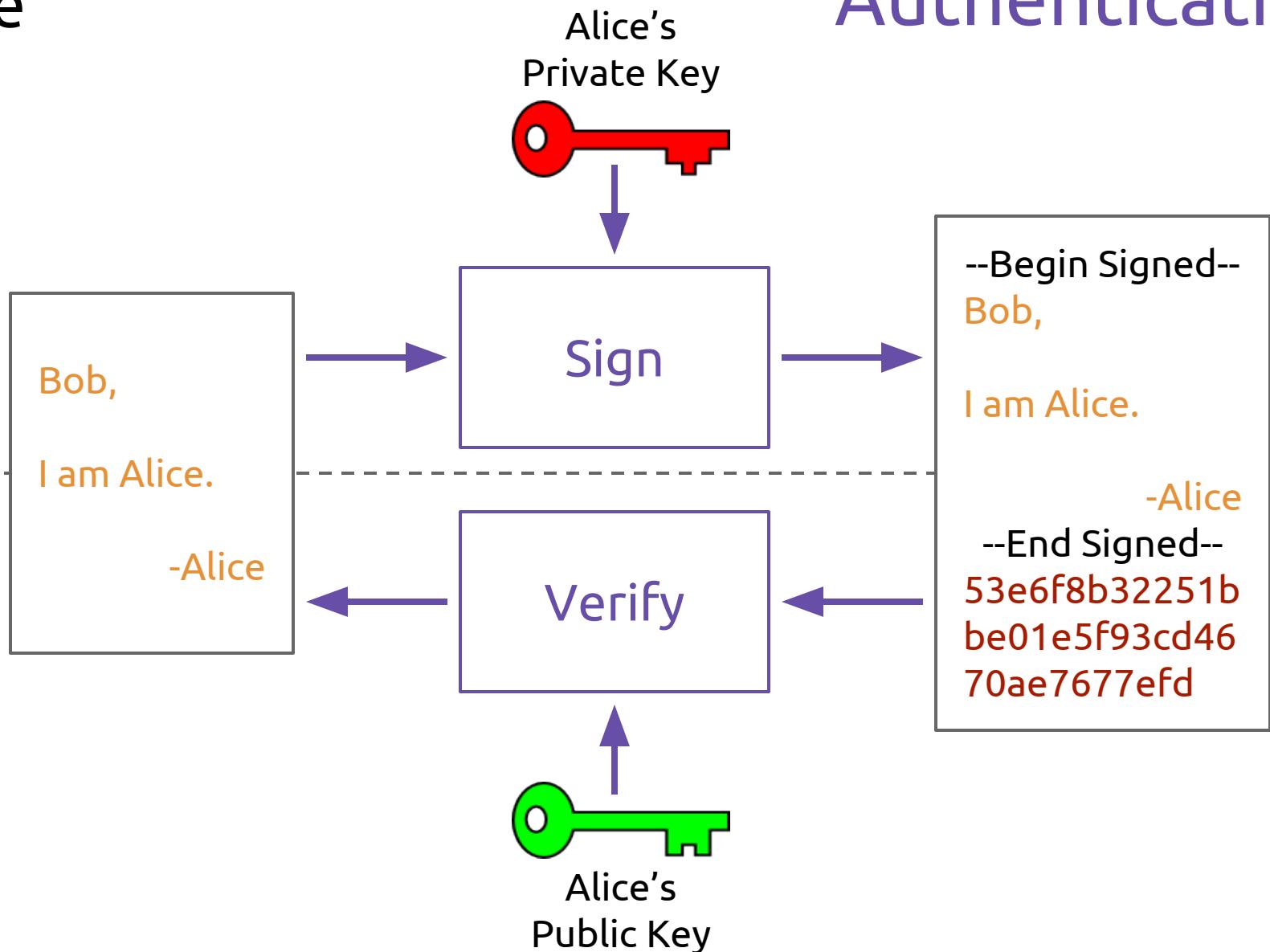
Signing



Bob

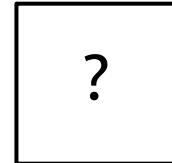
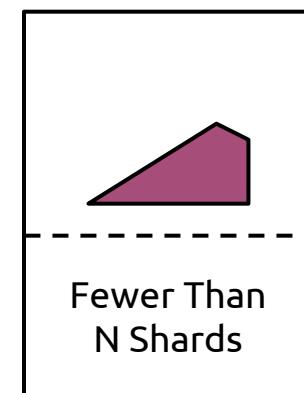
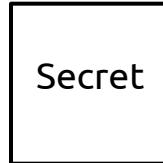
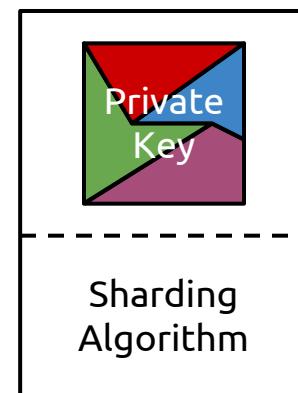
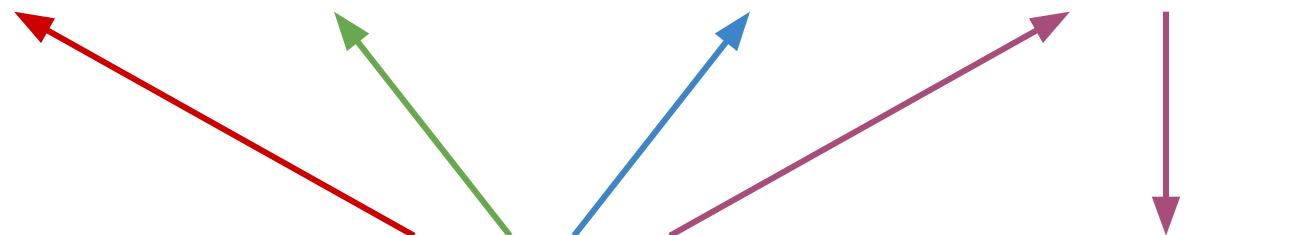
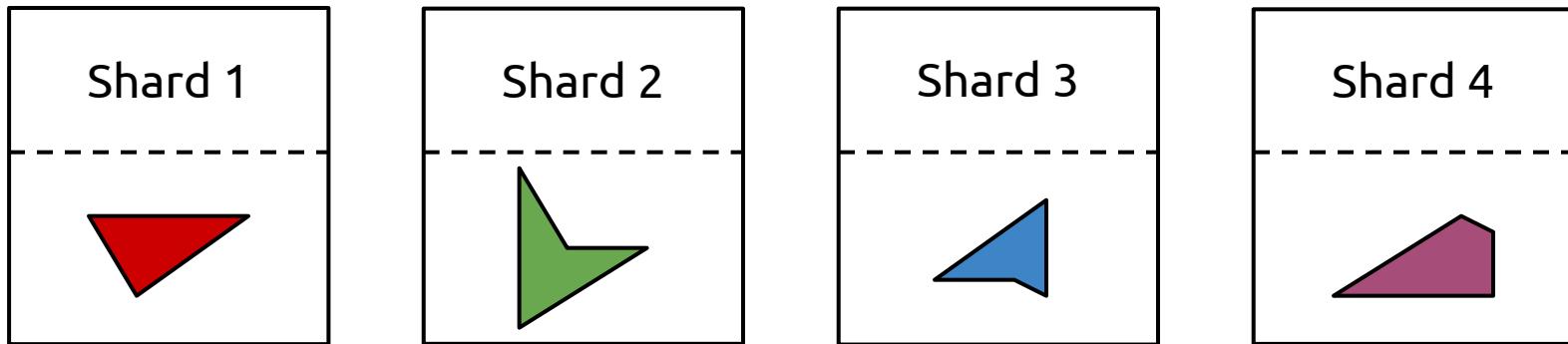
Alice

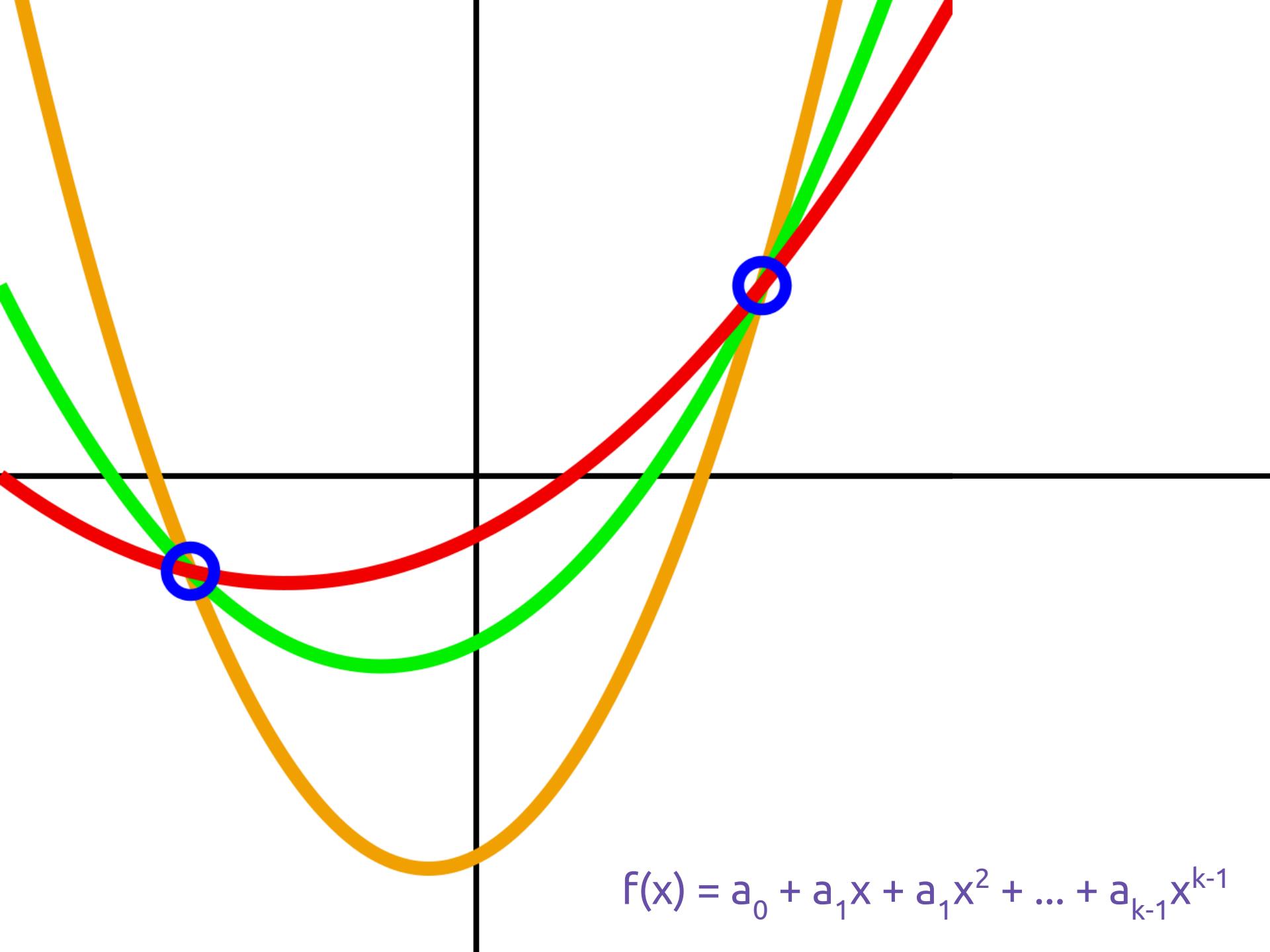
Authentication



Bob

Secret Sharing

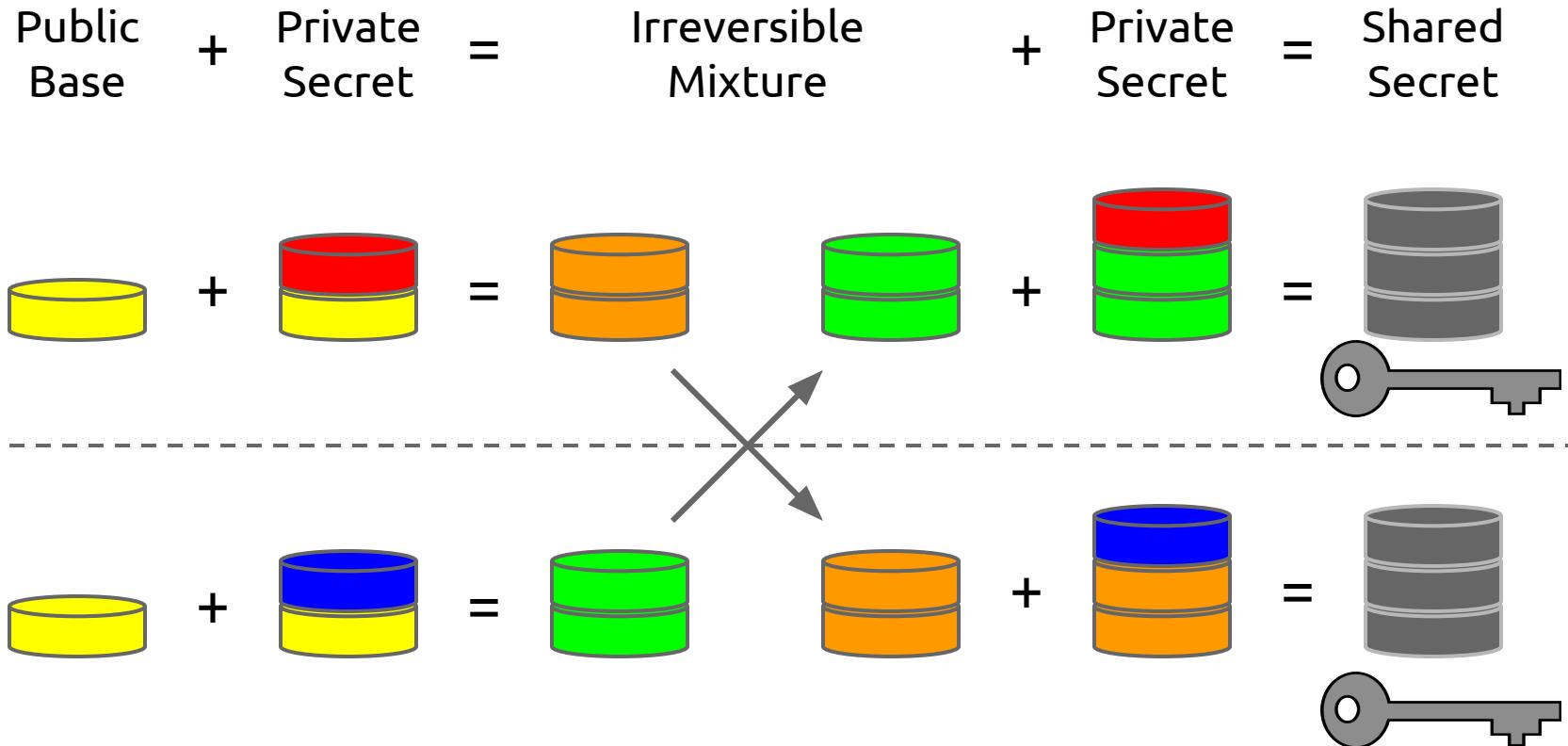




Diffie-Hellman

Alice

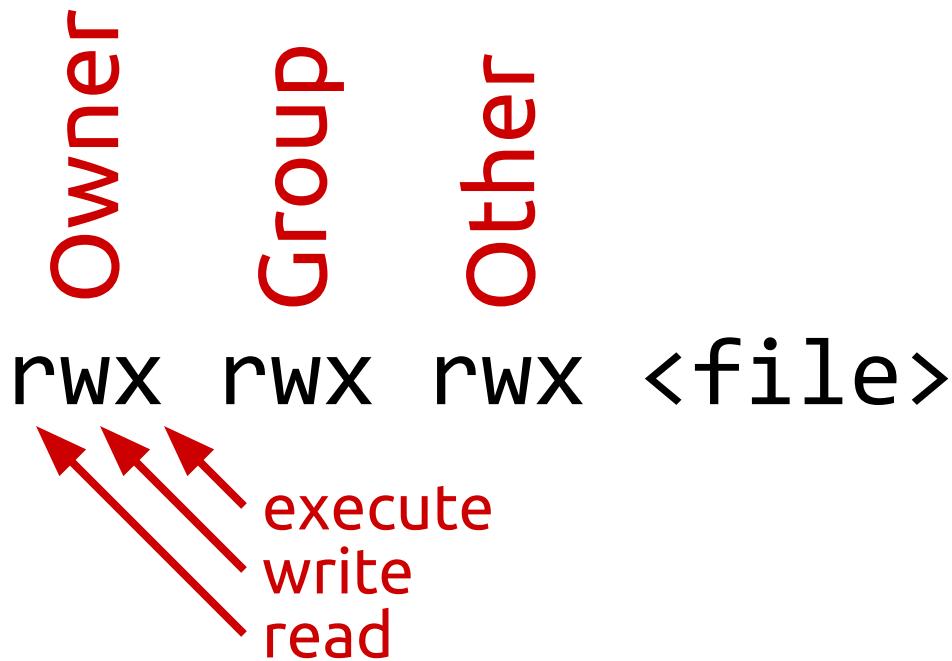
Key Exchange



Bob

Access Control

Unix Permissions



Windows NT ACLs

```
<file>
|--- read: (User A, User B)
|--- write: (User A, User B)
|--- delete: (User A)
|--- change perms: (User A)
|--- ...
```

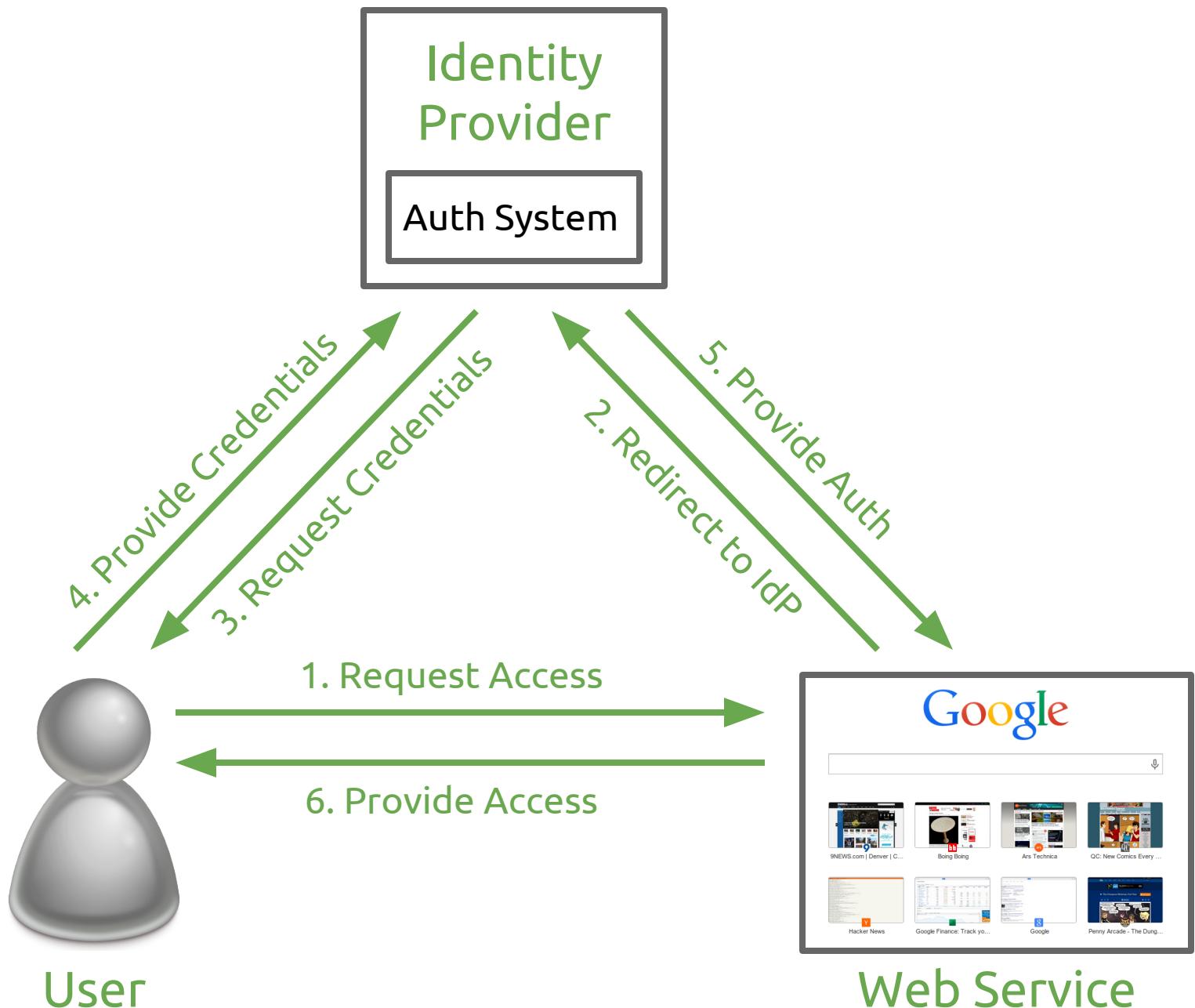
Roles

```
| --- Admin: (User A)
| --- Developer: (User A, User B)
| --- ...
```

<File>

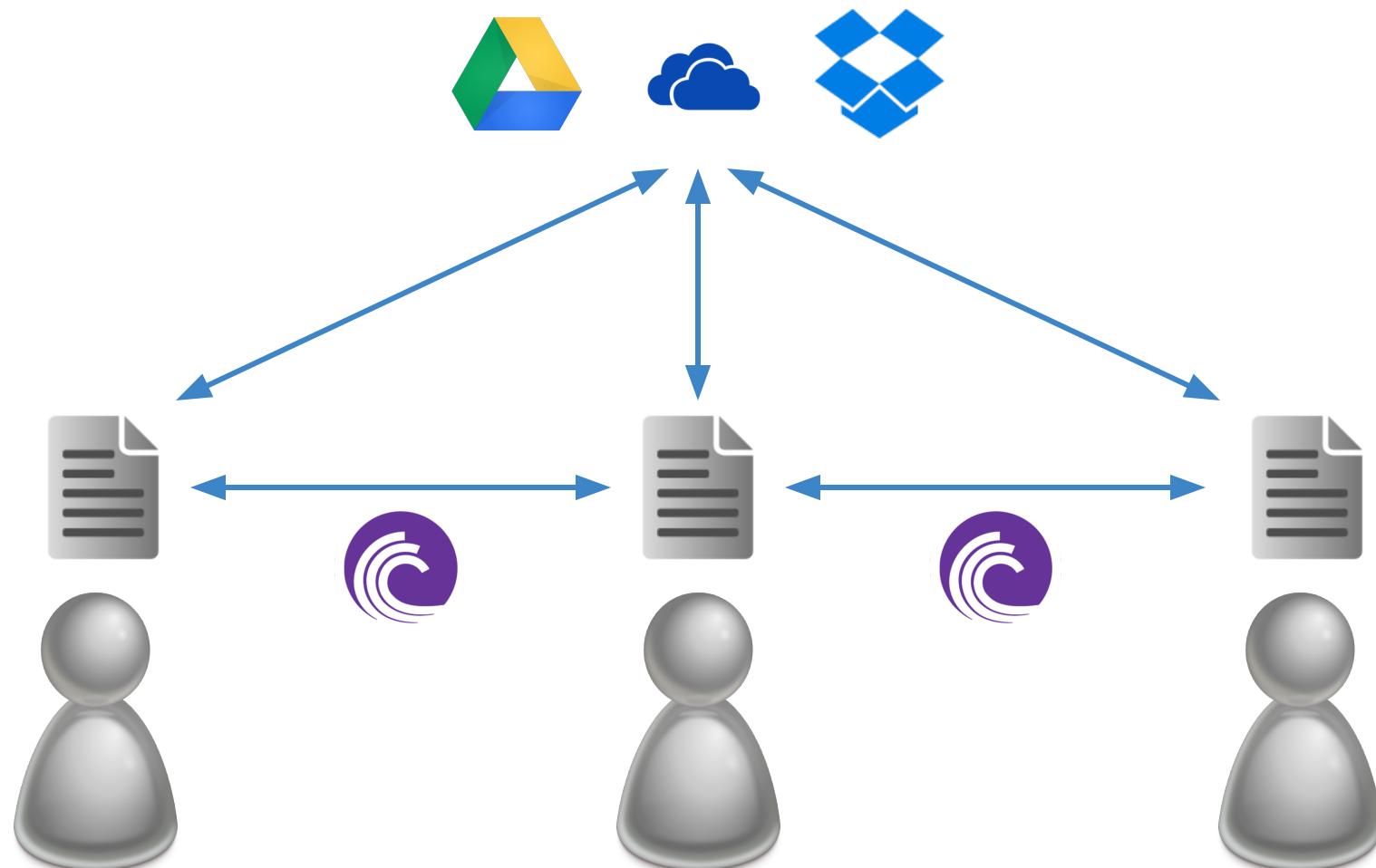
```
| --- read: (Admin, Developer)
| --- write: (Admin, Developer)
| --- delete: (Admin)
| --- ...
```

Federated Access Control

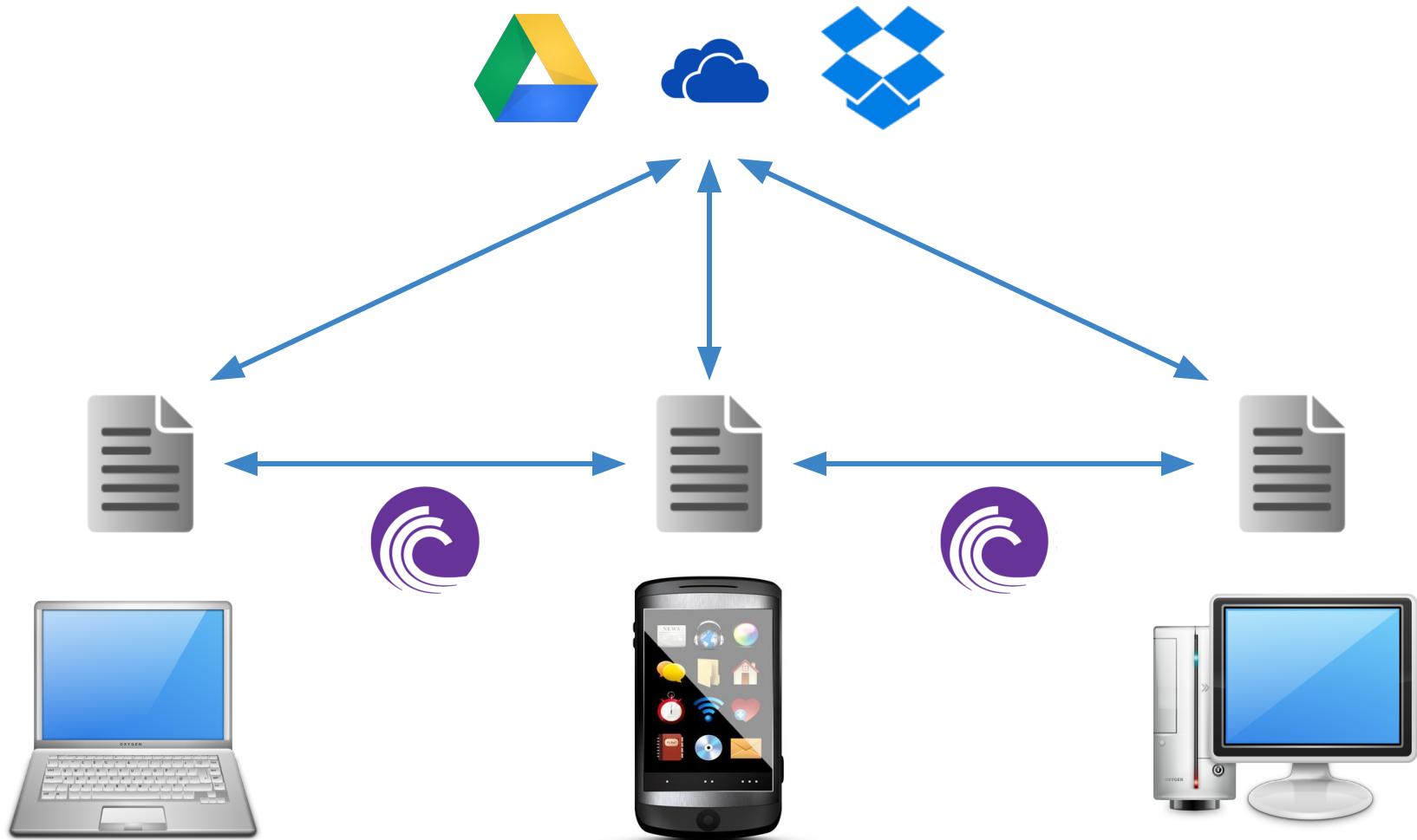


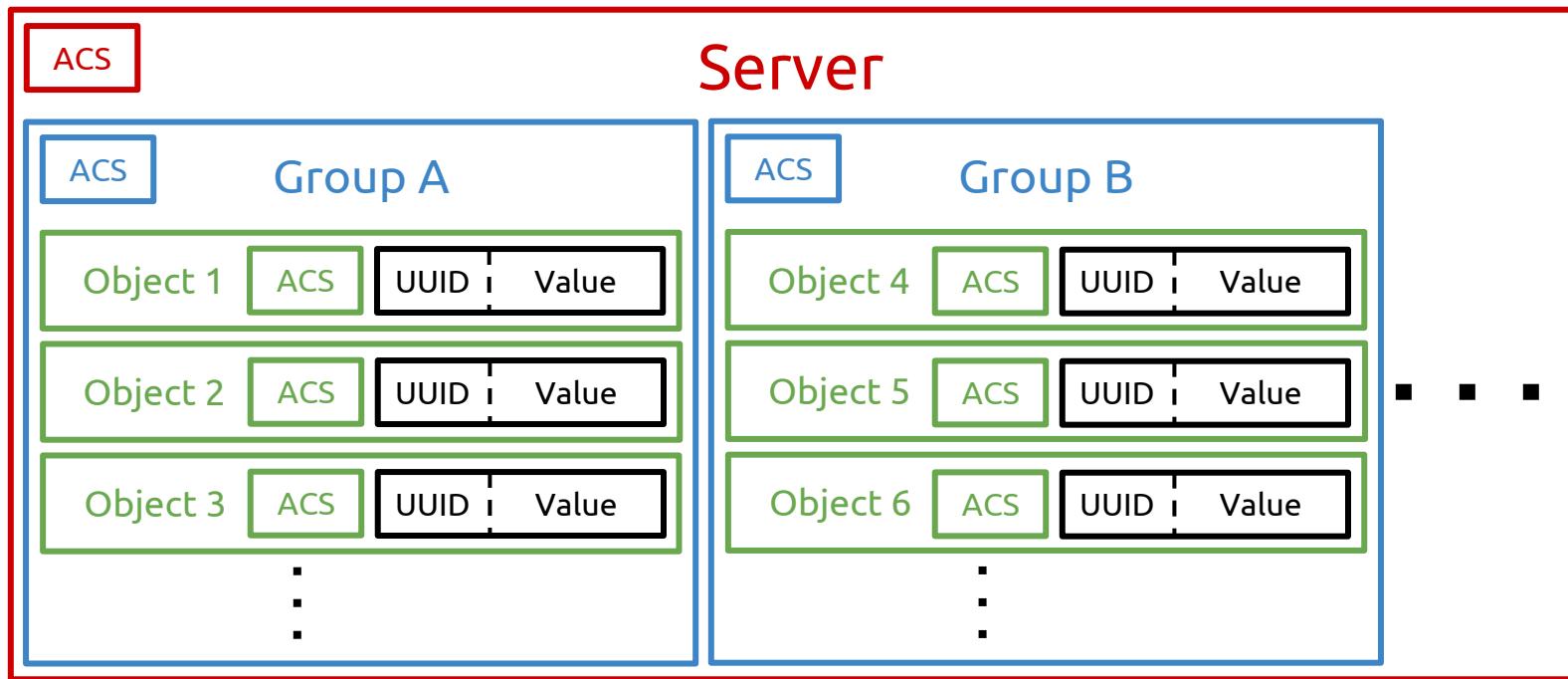
Custos Organizational Units

Multi-User File Sharing



Multi-Device File Access





Custos Access Control

Permissions

Permission	Rights
<code>srv_grp_create</code>	create groups on a Custos server
<code>srv_grp_list</code>	list groups on a Custos server
<code>srv_grp_override</code>	escalate to any group-level permission, overriding the per-group ACS
<code>srv_audit</code>	read all server-level audit information (i.e. group creation logging, group override logging, etc)
<code>srv_clean</code>	delete all server-level audit information (i.e. group creation logging, group override logging, etc)
<code>srv_acs_get</code>	view the server-level ACS controlling the permissions in this list
<code>srv_acs_set</code>	update the server-level ACS controlling the permissions in this list

Permission	Rights
<code>srv_grp_create</code>	create groups on a Custos server
<code>srv_grp_list</code>	list groups on a Custos server
<code>srv- grp- obj- create</code>	Permission
<code>srv- grp- obj- list</code>	Rights
<code>srv- grp- obj- override</code>	create a key:value objects within the given group
<code>srv- grp- delete</code>	list key:value objects within the given group
<code>srv- grp- audit</code>	escalate to any object-level permission, overriding the per-object ACS
<code>grp- clean</code>	delete the given group on a Custos server
<code>grp-acs- get</code>	read all group-level audit information (i.e. object creation logging, object override logging, etc)
<code>grp-acs- set</code>	delete all group-level audit information (i.e. object creation logging, object override logging, etc)
<code>grp-acs- get</code>	view the group-level ACS controlling the permissions in this list
<code>grp-acs- set</code>	update the group-level ACS controlling the permissions in this list

Permission	Rights
<code>srv_grp_create</code>	create groups on a Custos server
<code>srv_grp_list</code>	list groups on a Custos server
<code>srv- grp- obj- audit- clean- acs-</code>	Permission
<code>srv- grp- obj- audit- clean- acs-</code>	Rights
<code>grp_obj_create</code>	create a key:value objects within the given group
<code>grp_obj_list</code>	list key:value objects within the given group
<code>srv- grp- obj- audit- clean- acs-</code>	Permission
<code>srv- grp- obj- audit- clean- acs-</code>	Rights
<code>obj_delete</code>	delete the given key:value object within the given group
<code>obj_read</code>	read the given key:value object within the given group
<code>obj_update</code>	create a new version of the given key:value object within the given group (the equivalent of a “write” permission for the Custos write-once system)
<code>obj_audit</code>	read all object-level audit information (i.e. object read logging, object update logging, etc)
<code>obj_clean</code>	delete all object-level audit information (i.e. object read logging, object update logging, etc)
<code>obj_acs_get</code>	view the object-level ACS controlling the permissions in this list
<code>obj_acs_set</code>	update the object-level ACS controlling the permissions in this list

Authentication Attributes

Authentication Attributes

Plugin-Based

Explicit

ip_src

user_agent

time_utc

...

Implicit

user_id

psk

psk_sha256

...

Access Control Chain

```
[  
  [ (username = 'Andy'),  
    (password = '12345'),  
    (src_ip = 192.168.1.0/24) ],  
  [ (username = 'Andy'),  
    (password = '12345'),  
    (src_ip = 75.148.118.216/29) ],  
  [ (username = 'John'),  
    (password = 'Swordfish') ]  
]
```

```
(username = 'Andy')
|
(password = '12345')
/
(src_ip = 192.168.1.0/24) (src_ip = 75.148.118.216/29)
```

```
(username = 'John')
|
(password = 'Swordfish')
```

“Secret”

“Secret”

Access Control Specification (ACS)

“Secret”

Access Control Specification (ACS)

Permission A

“Secret”

Access Control Specification (ACS)

Permission A

Access Control
Chain

“Secret”

Access Control Specification (ACS)

Permission A

Access Control
Chain

Auth
Attribute



Auth
Attribute



Auth
Attribute

“Secret”

Access Control Specification (ACS)

Read Permission

Access Control
Chain

Auth
Attribute



Auth
Attribute



Auth
Attribute

“Secret”

Access Control Specification (ACS)

Read Permission

Access Control
Chain

Username



IP Address



Password

“Secret”

Access Control Specification (ACS)

Read Permission

Access Control
Chain

Username



IP Address



Password

Update Perm.

Access Control
Chain

Username

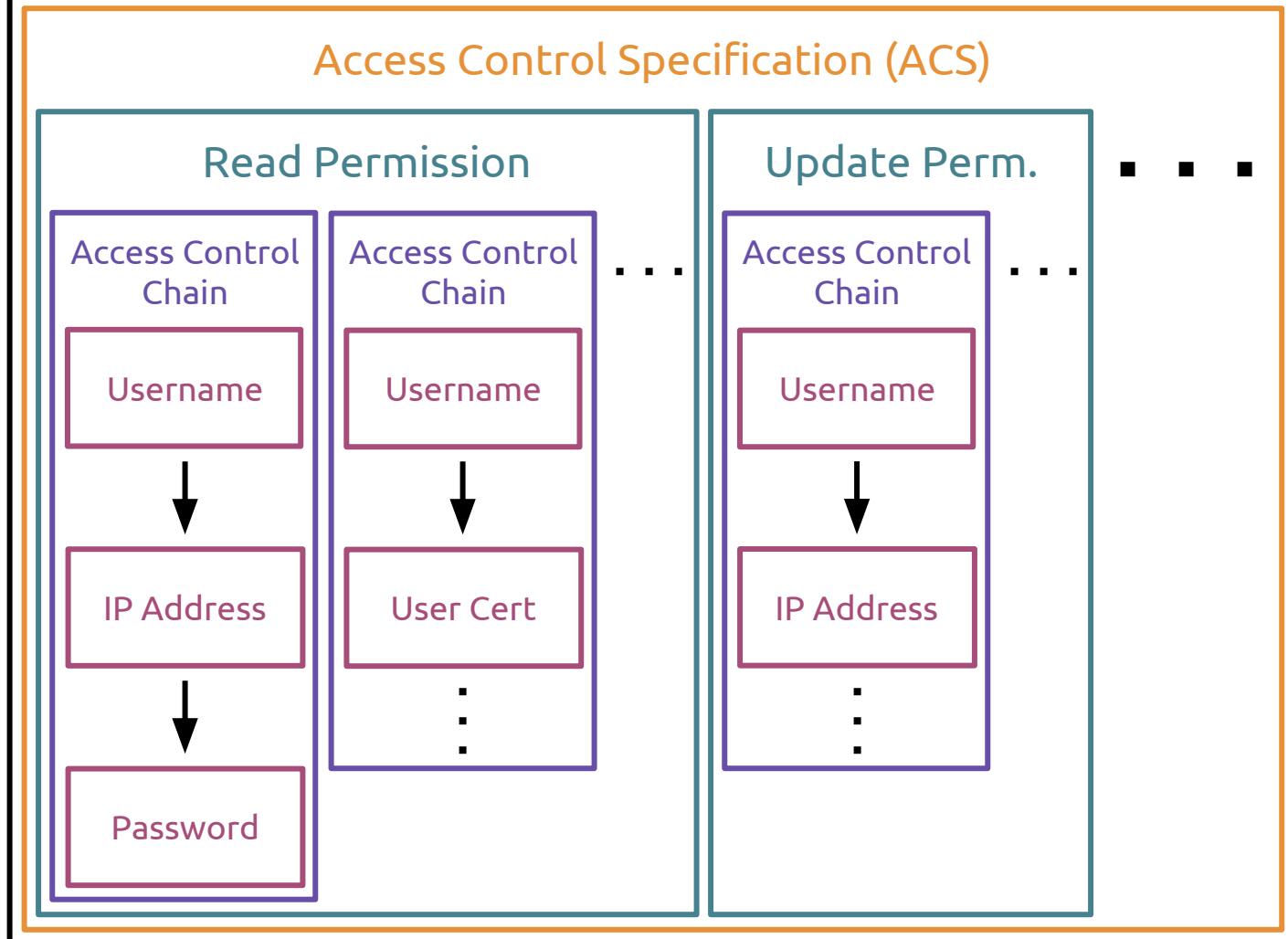


IP Address



■ ■ ■

“Secret”

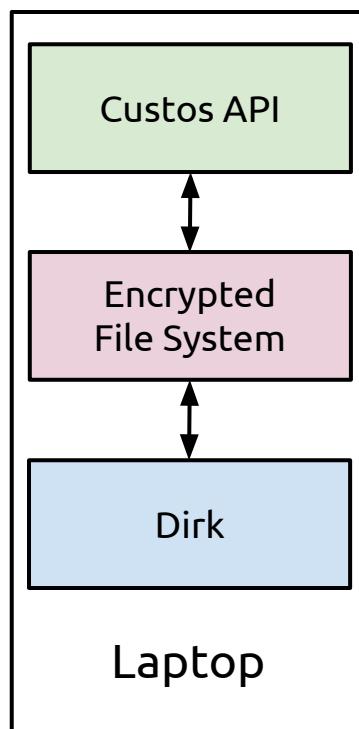
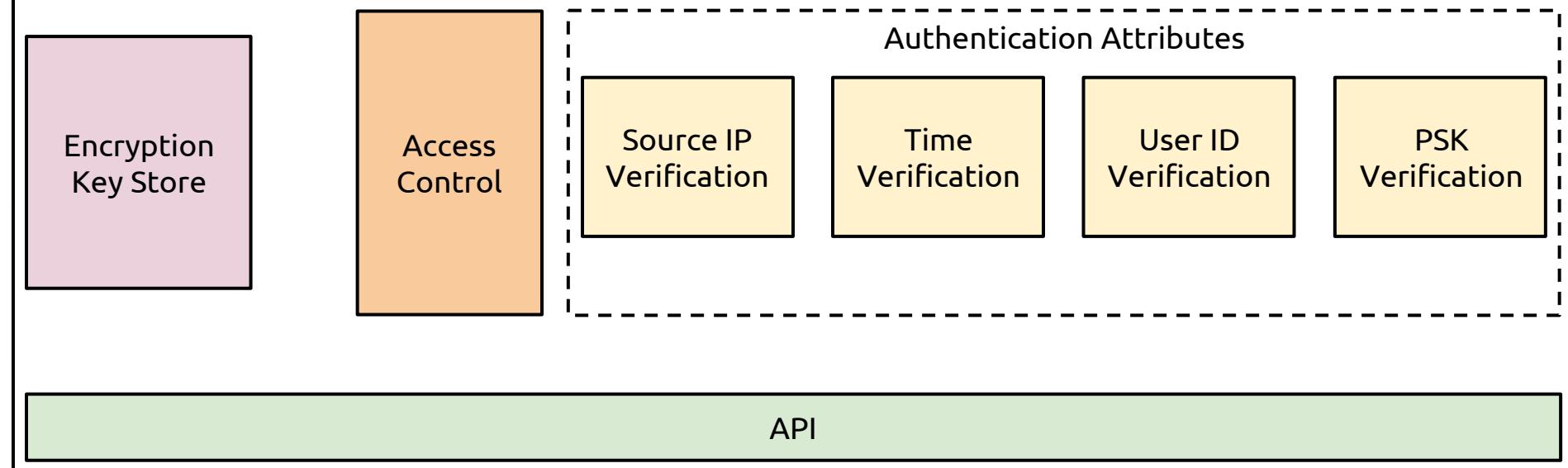


Access Example

619a06f0-50af-11e3-8f96-0800200c9a66 ACS

```
{  
    obj_read:  
        [  
            [ (ip\src = '1.2.3.4'),  
             (time\utc = '1300 +/- 5') ],  
            [ (user\id = 'Dirk'),  
              (psk = 'ImaHakzor') ]  
            ...  
        ]  
        ...  
}
```

Custos Server



Request:

619a06f0-50af-11e3-8f96-0800200c9a66

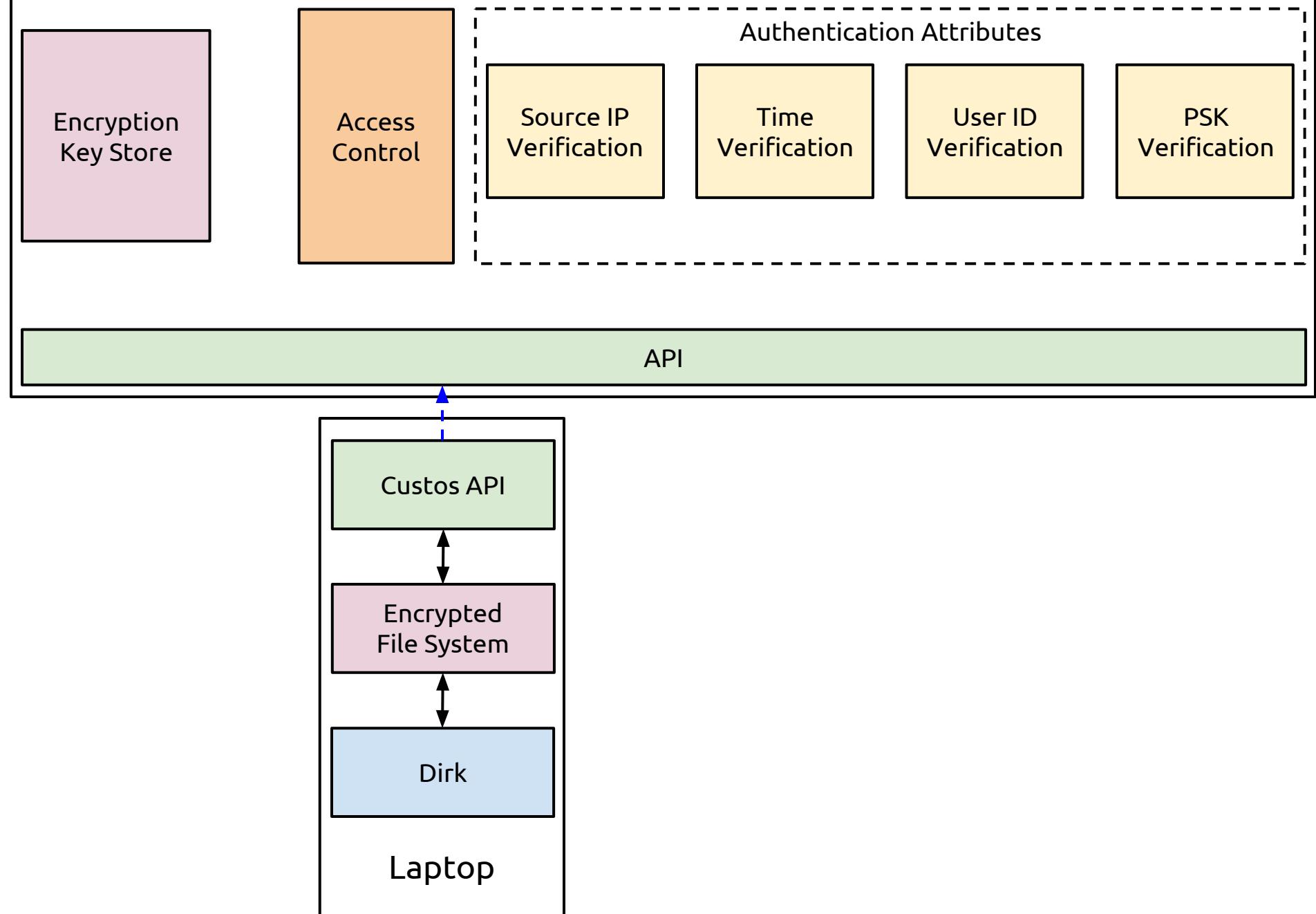
Authentication Attributes:

user_id = Dirk

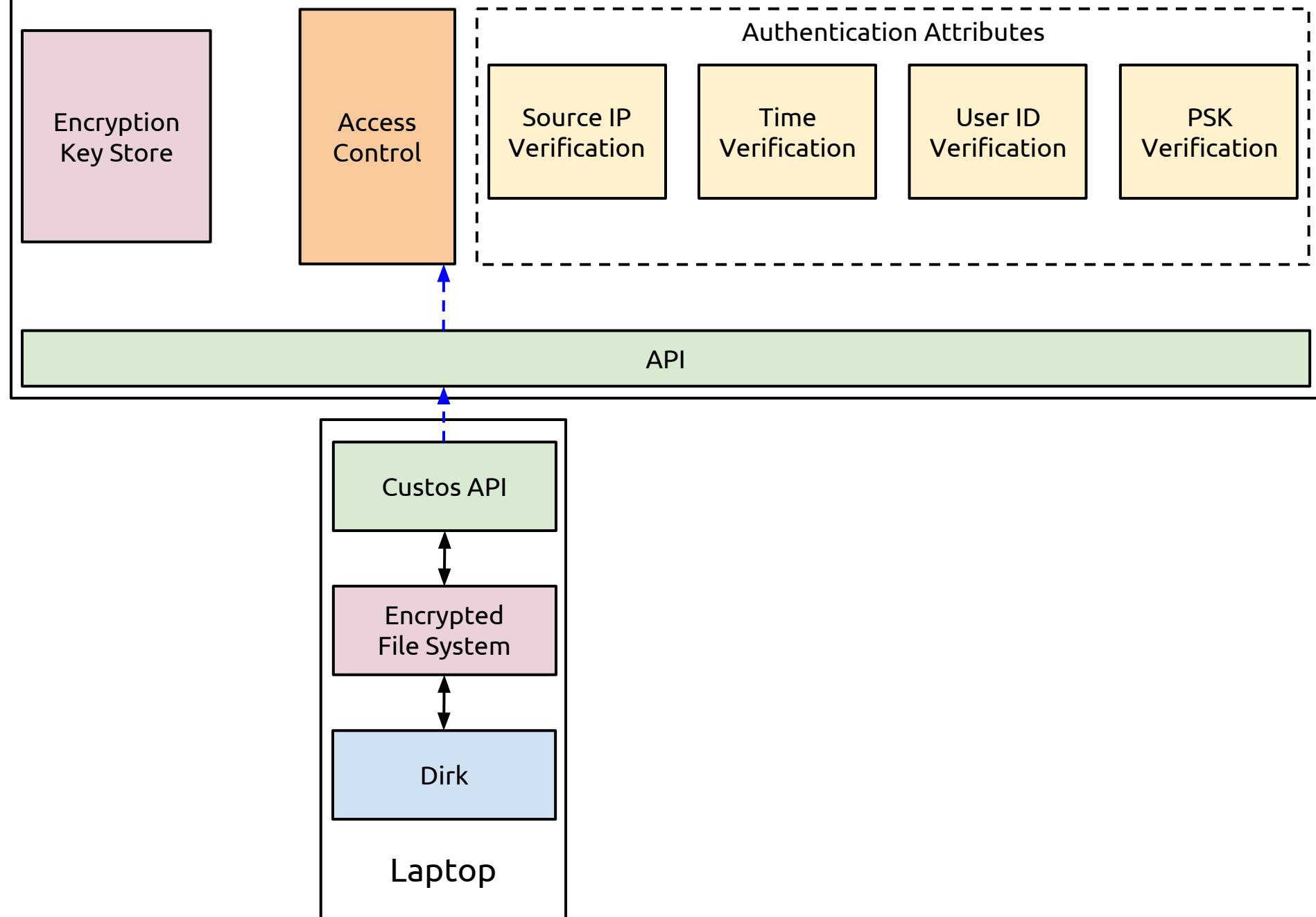
(ip_src = '1.2.3.4')

(time_utc = '1133')

Custos Server

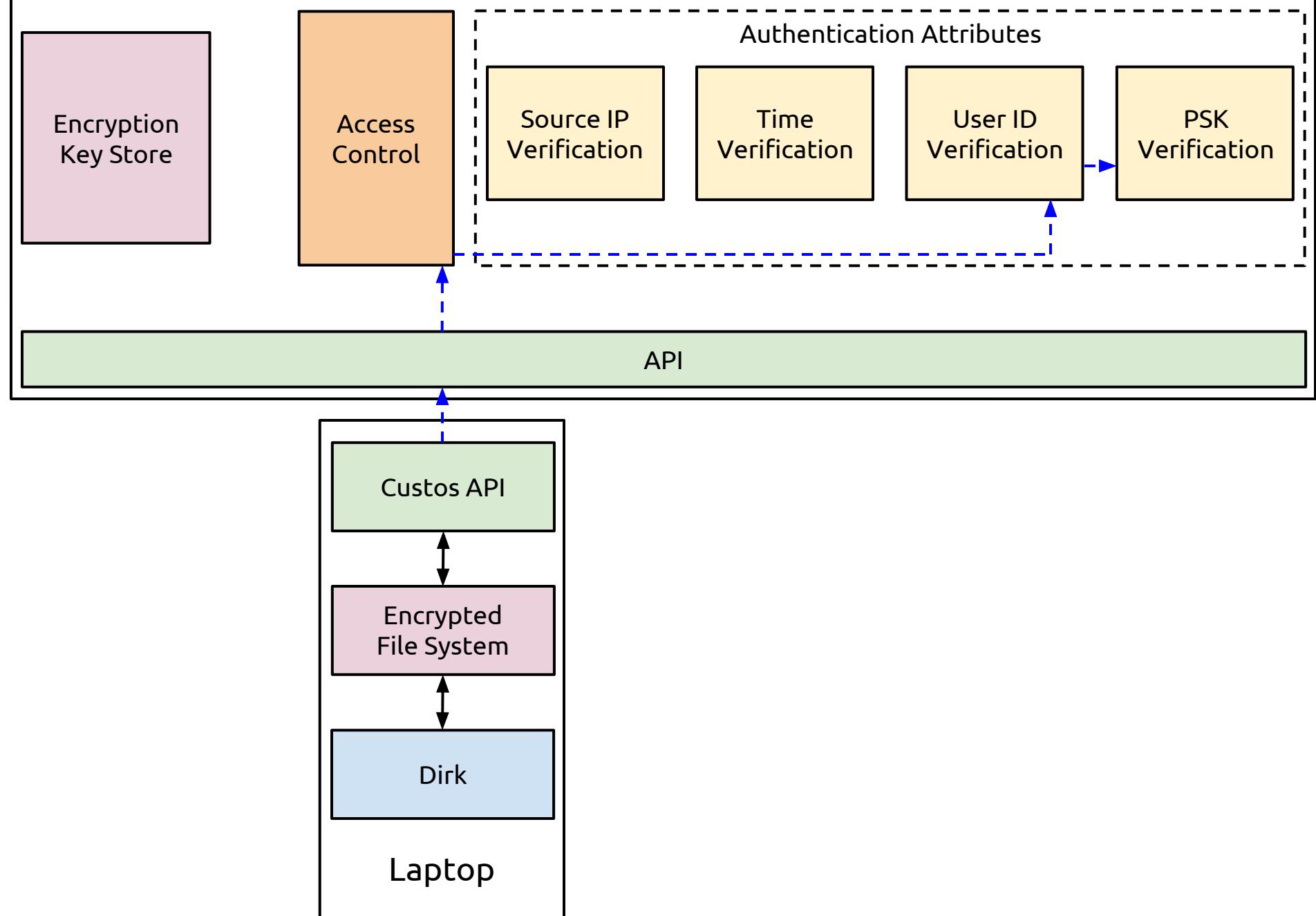


Custos Server



```
{  
    obj_read:  
        [  
            [ (ip\src = '1.2.3.4'),  
              (time\utc = '1300 +/- 5') ],  
            [ (user\id = 'Dirk'),  
              (psk = 'ImaHakzor') ]  
            . . .  
        ]  
        . . .  
}
```

Custos Server



Request:

619a06f0-50af-11e3-8f96-0800200c9a66

Authentication Attributes:

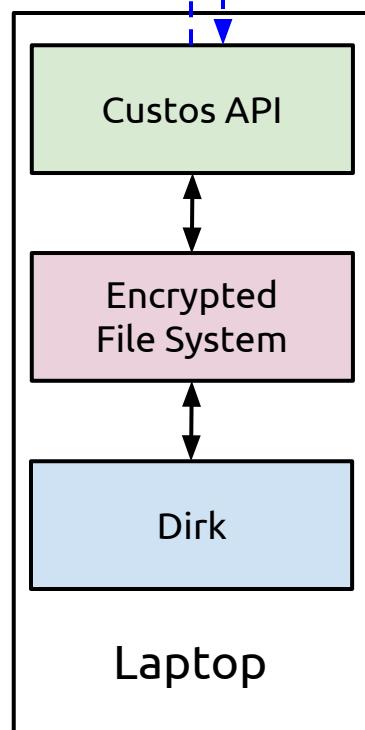
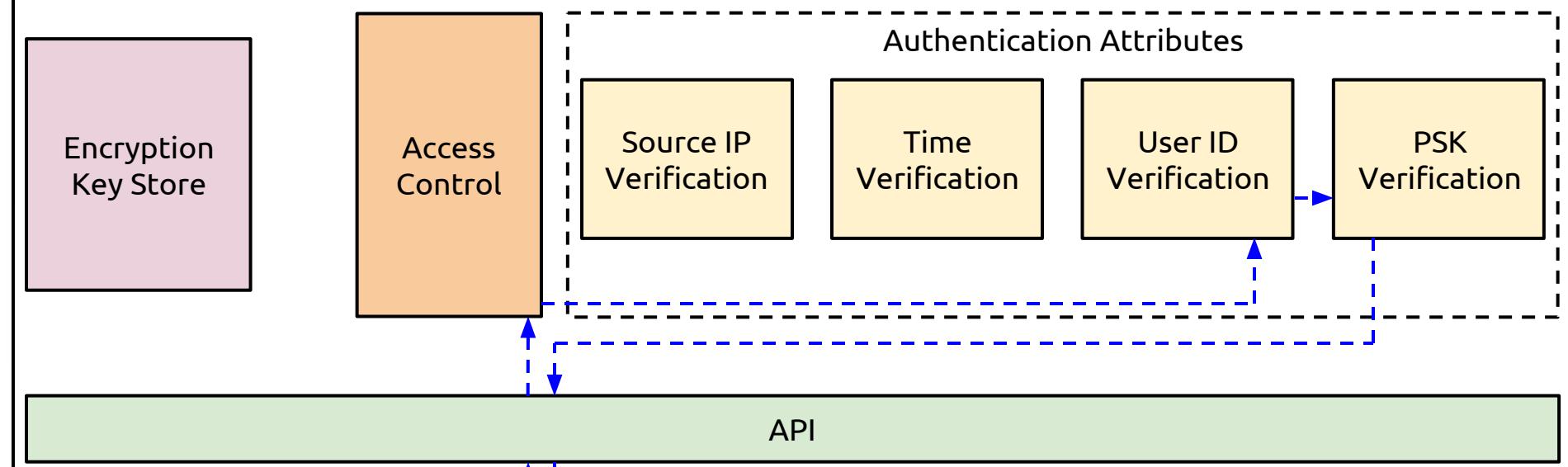
user_id = Dirk

(ip_src = '1.2.3.4')

(time_utc = '1133')

```
{  
    obj_read:  
        [  
            [ (ip\src = '1.2.3.4'),  
              (time\utc = '1300 +/- 5') ],  
            [ (user\id = 'Dirk'),  
              (psk = 'ImaHakzor') ]  
            . . .  
        ]  
        . . .  
}
```

Custos Server



Request:

619a06f0-50af-11e3-8f96-0800200c9a66

Authentication Attributes:

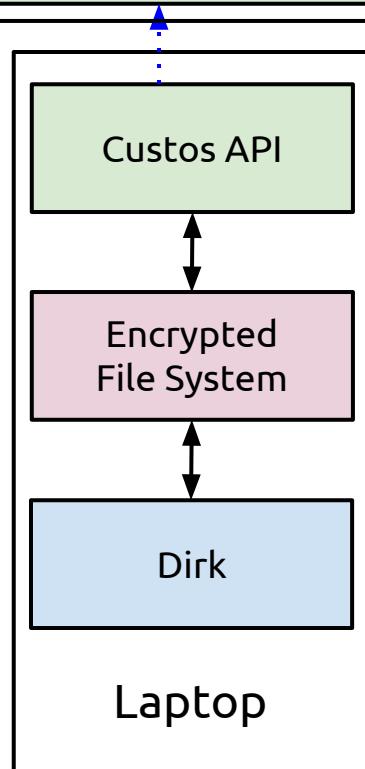
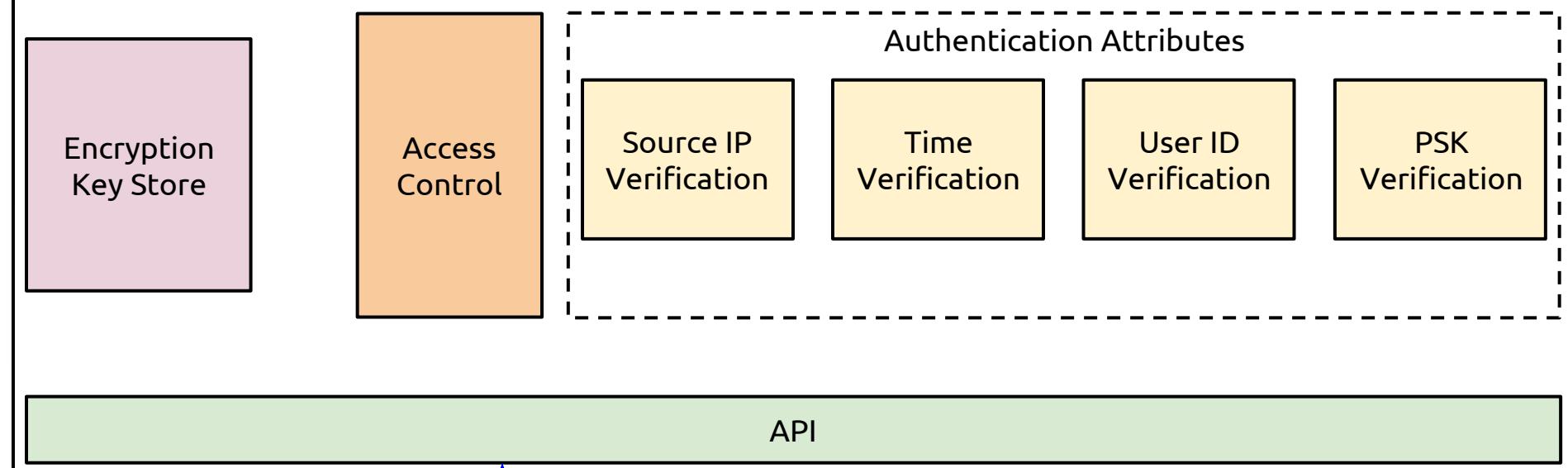
user_id = ‘Dirk’

psk = ‘ImaHackzor’

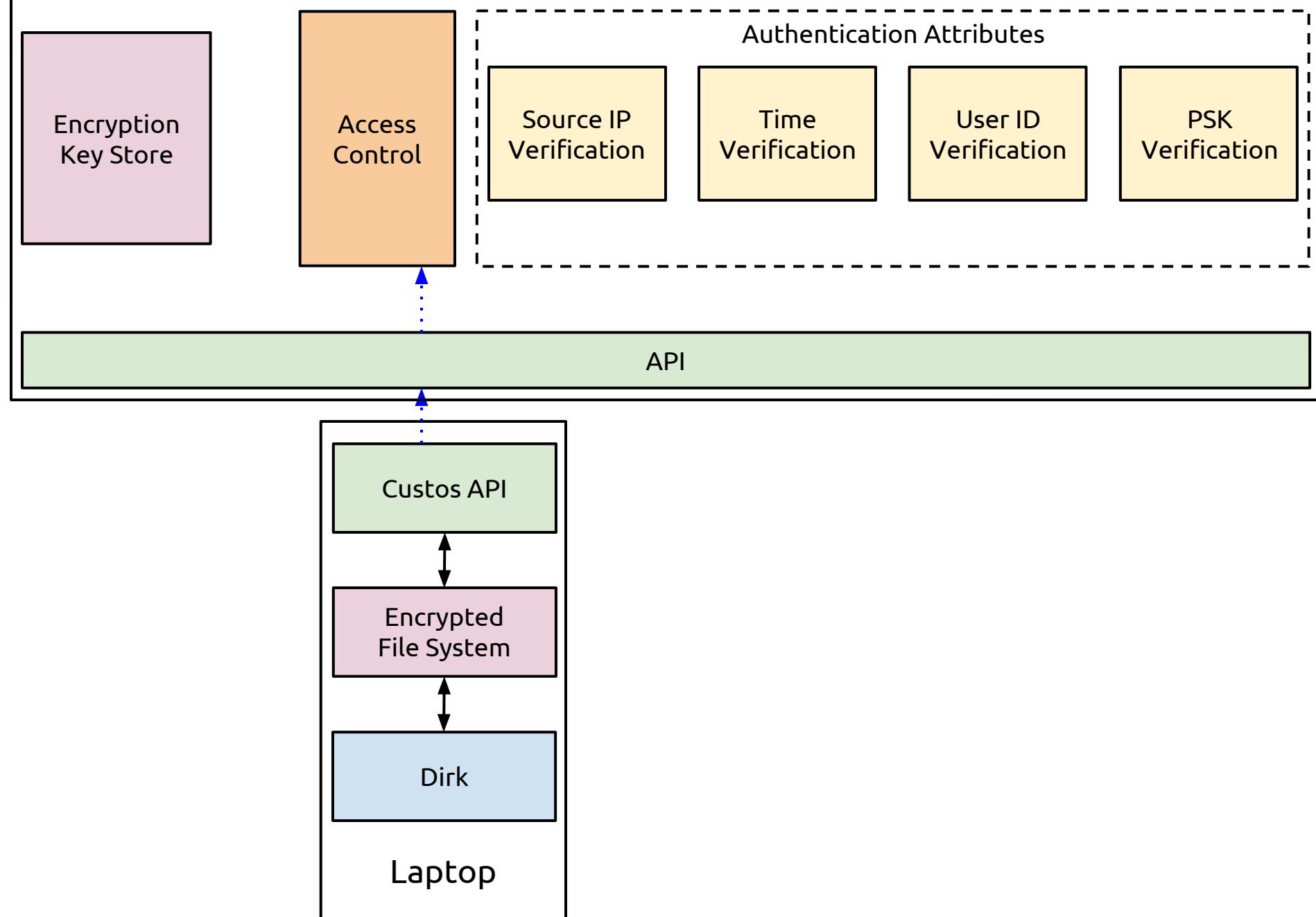
(ip_src = ‘1.2.3.4’)

(time_utc = ‘1133’)

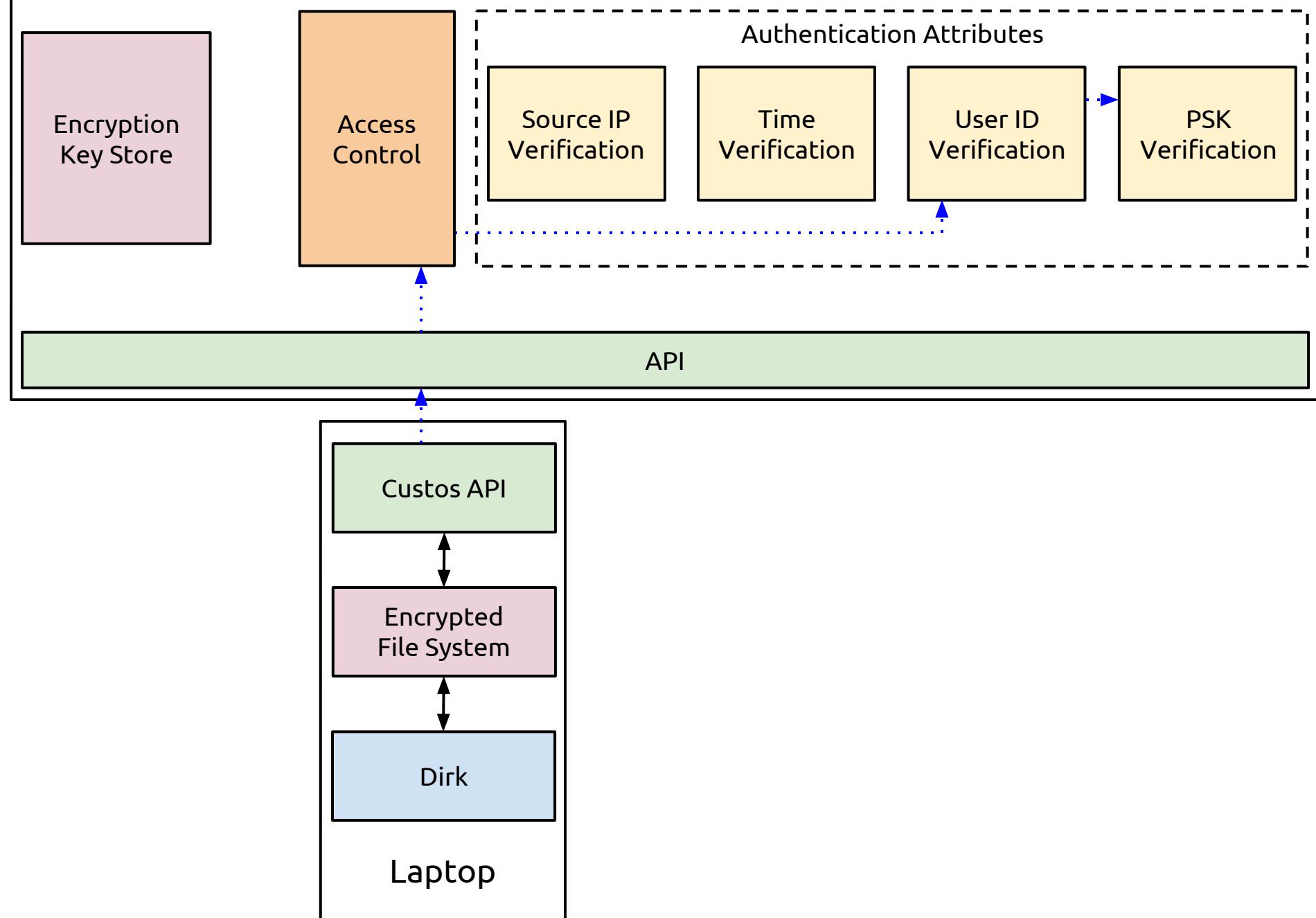
Custos Server



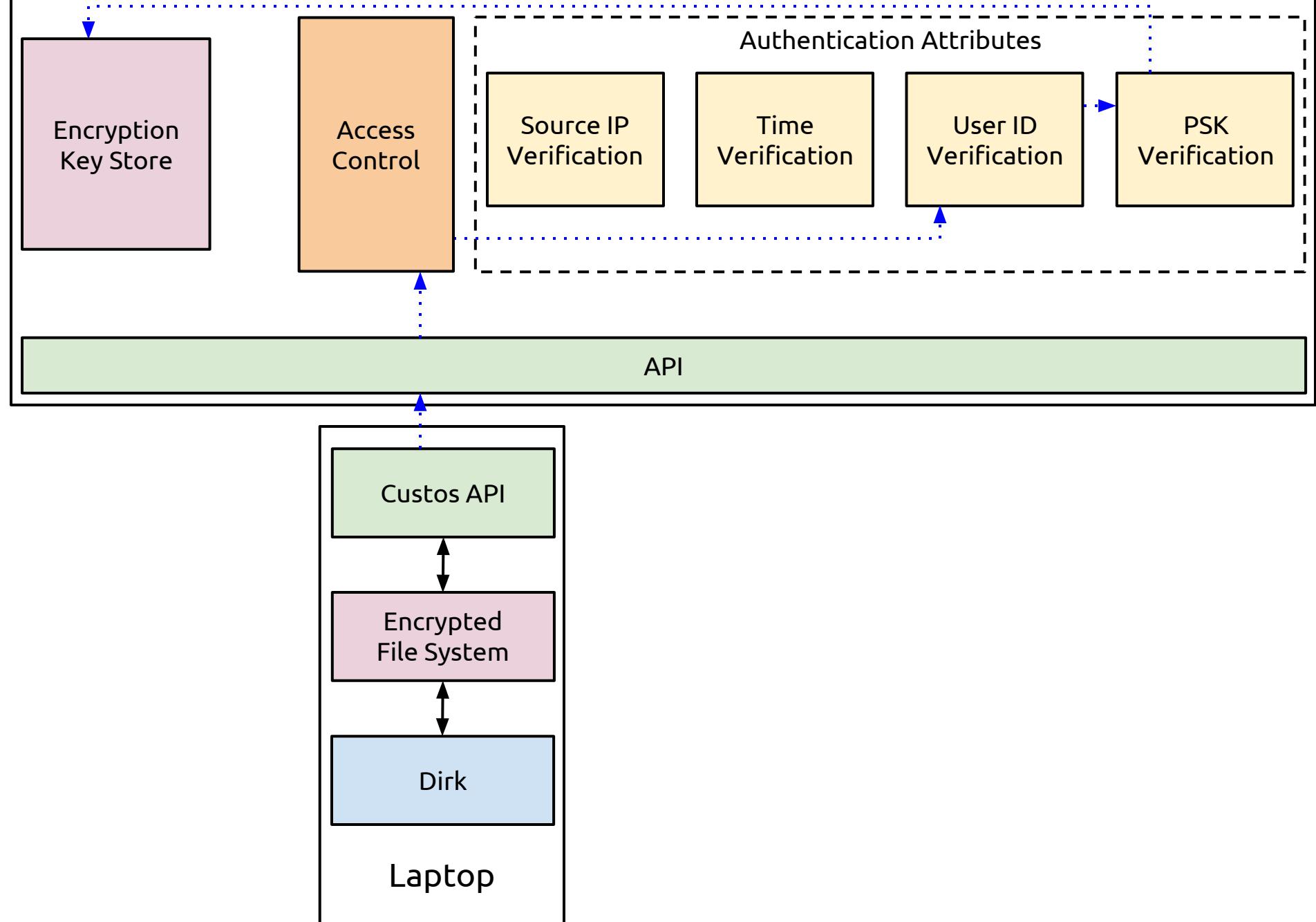
Custos Server



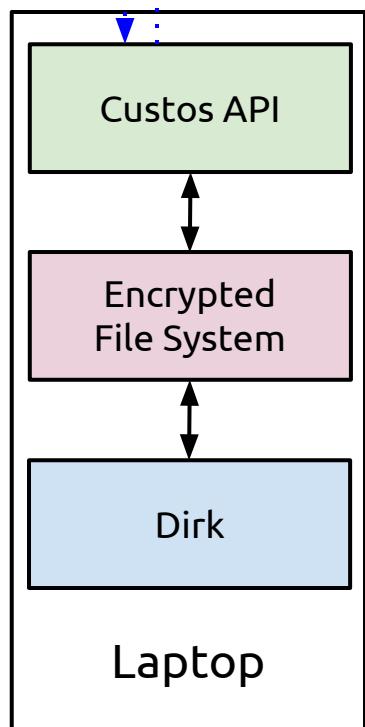
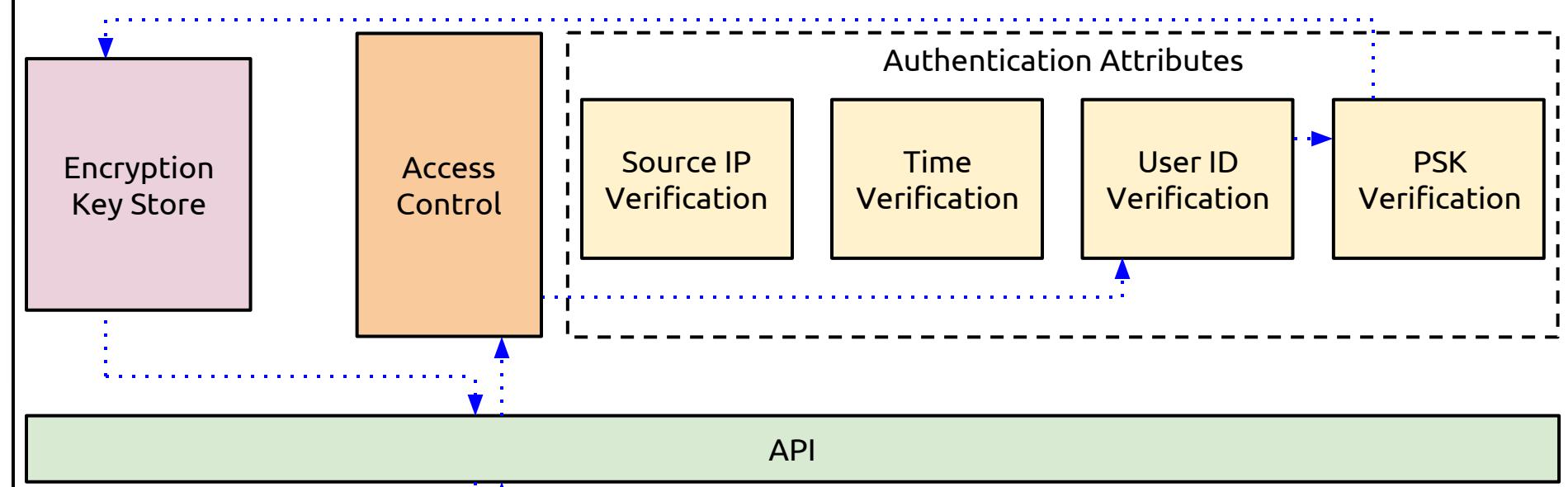
Custos Server



Custos Server

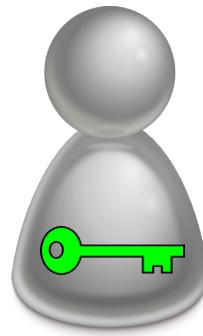


Custos Server



Revoking Access

Example: Revoke Shared Access



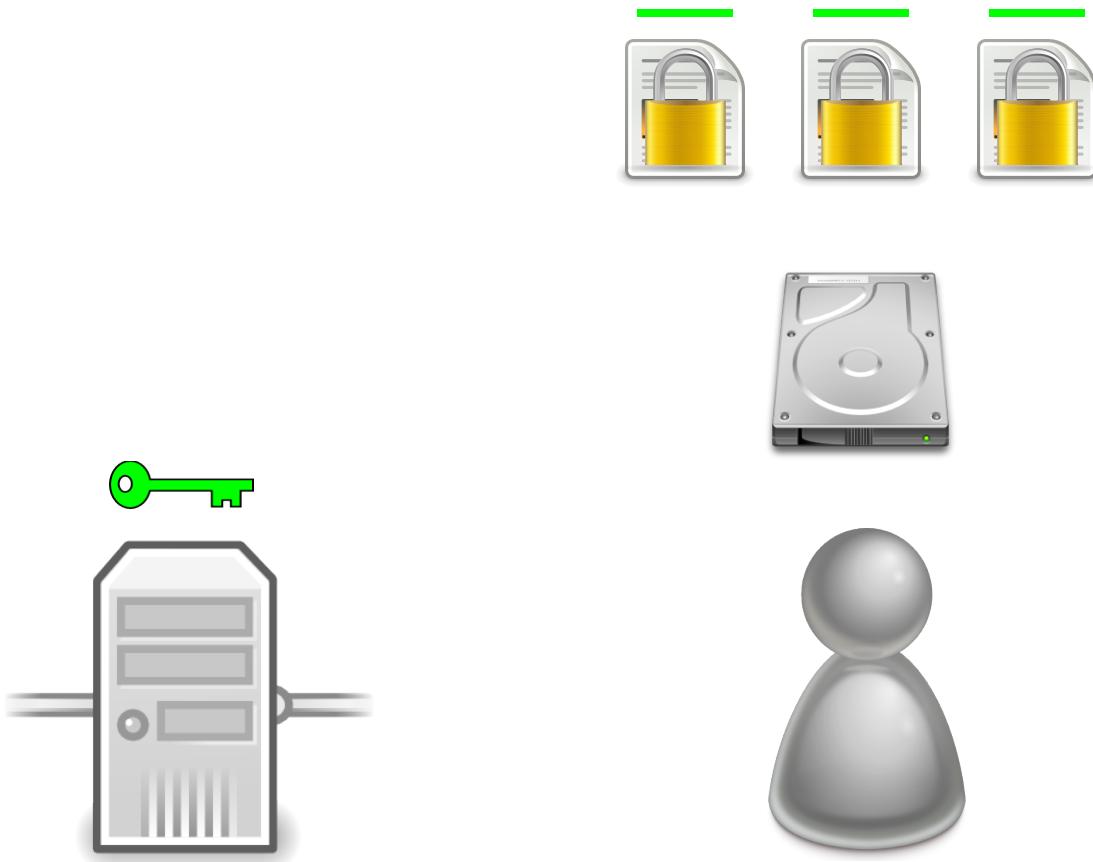
Example: Revoke Shared Access



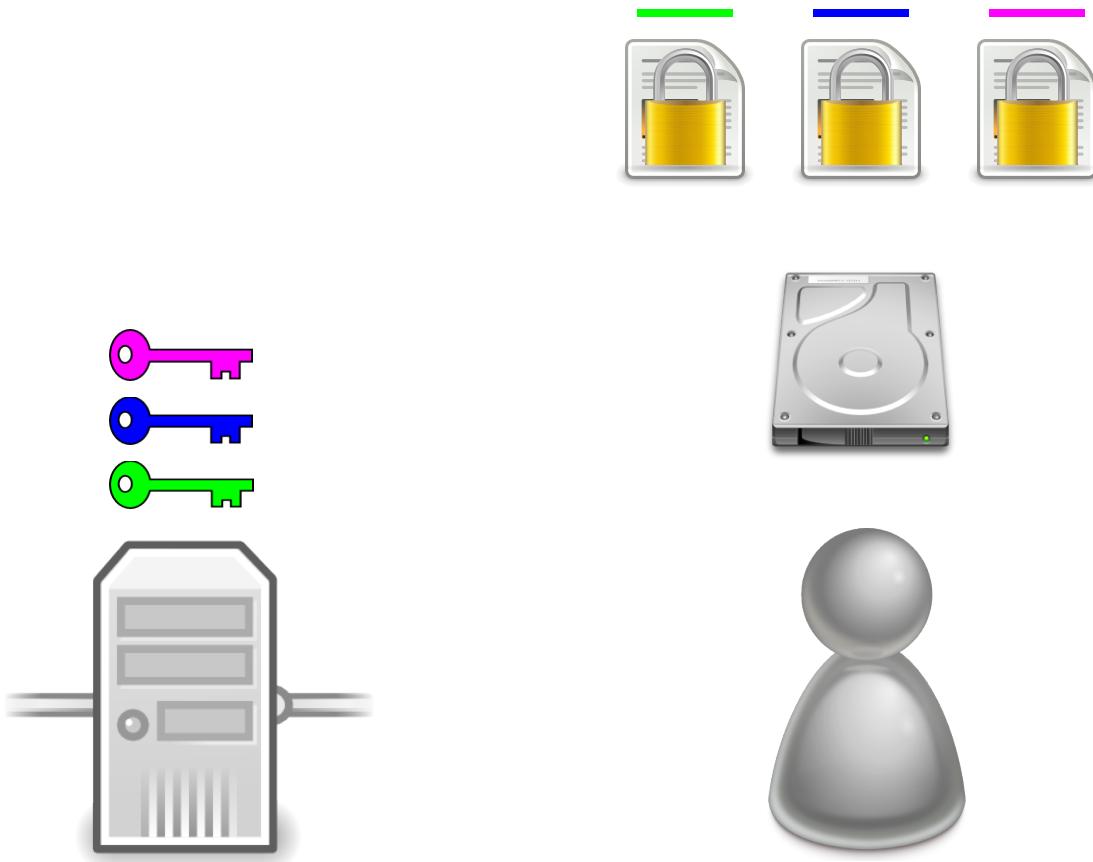
Example: Revoke Shared Access



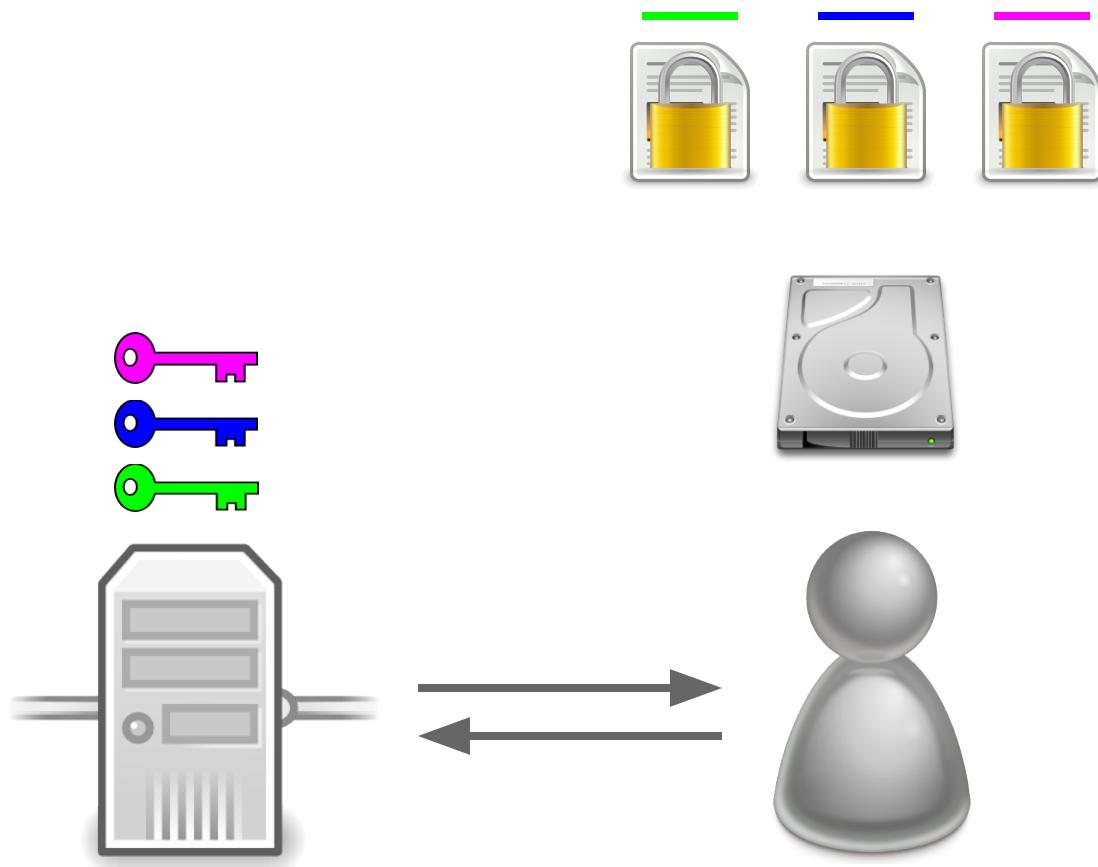
Example: Revoke Shared Access



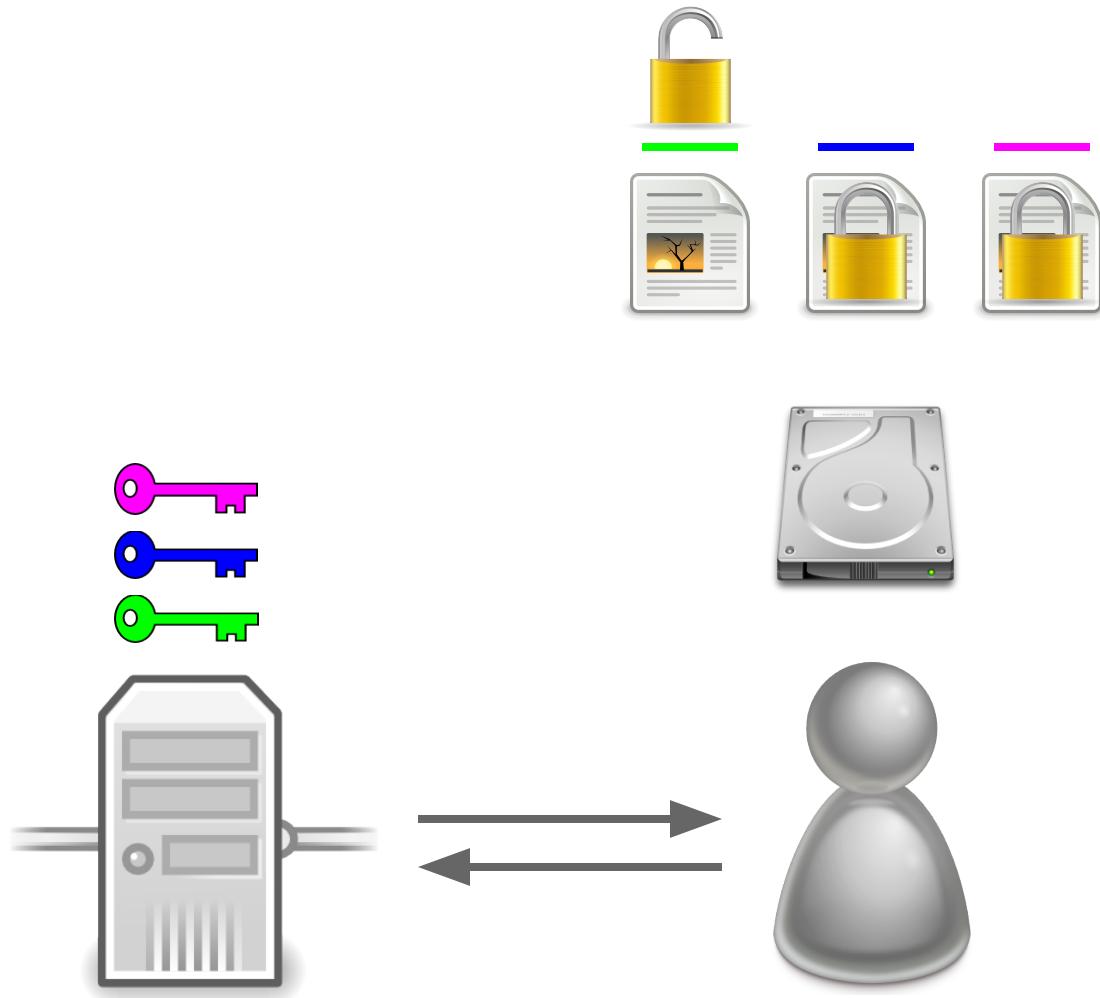
Example: Revoke Shared Access



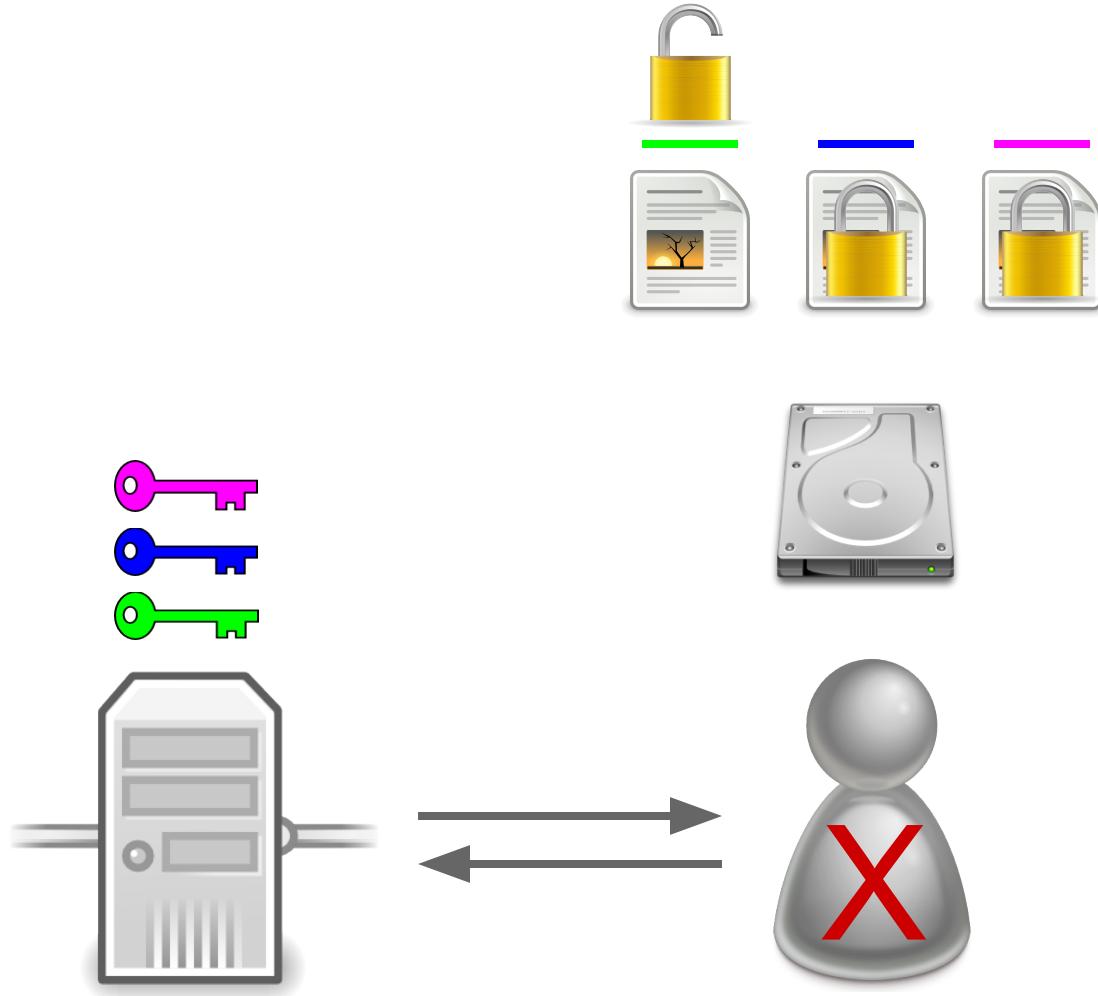
Example: Revoke Shared Access



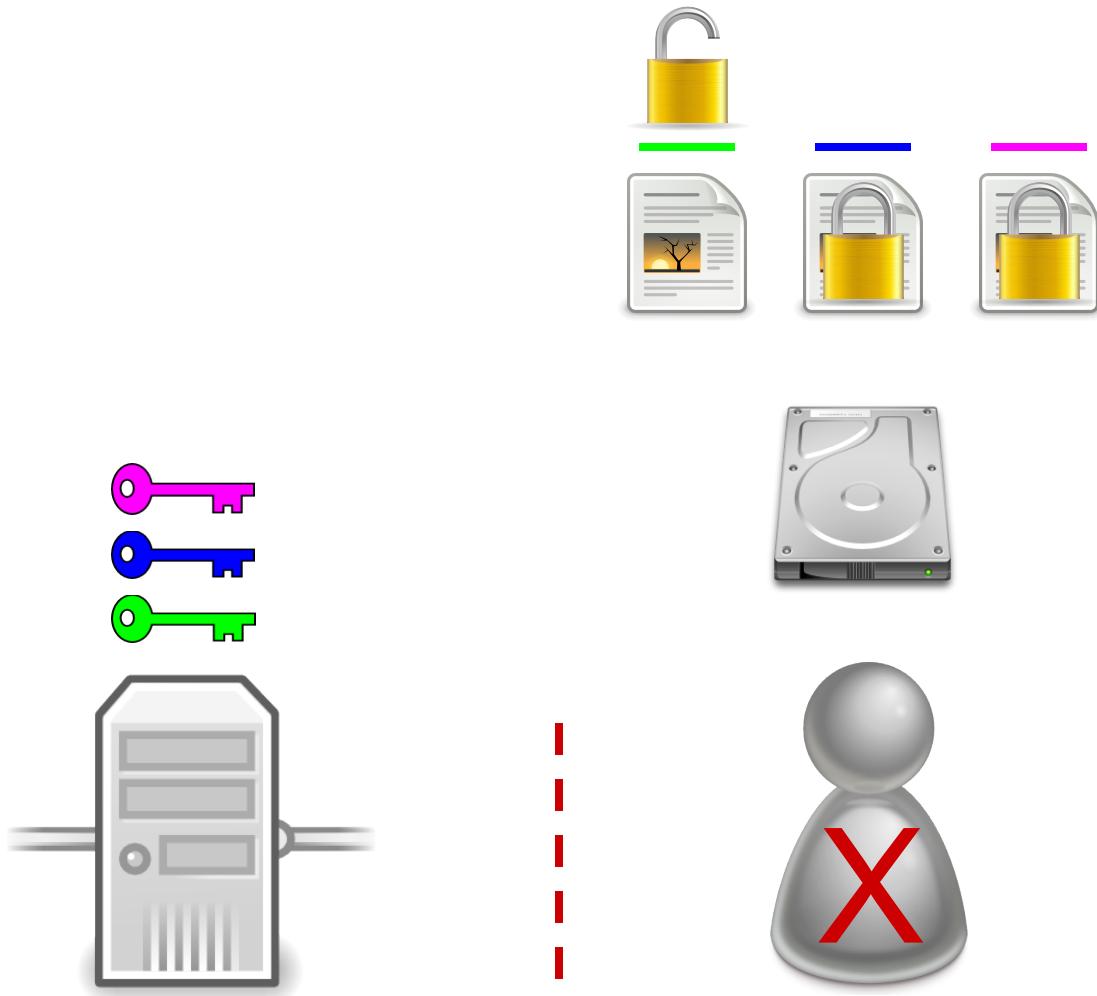
Example: Revoke Shared Access



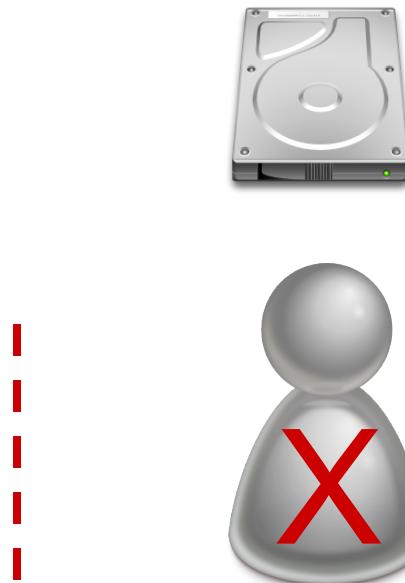
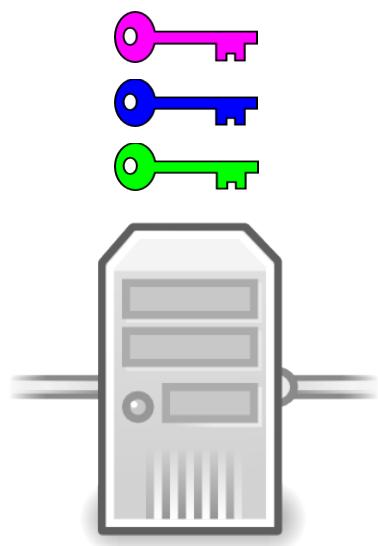
Example: Revoke Shared Access



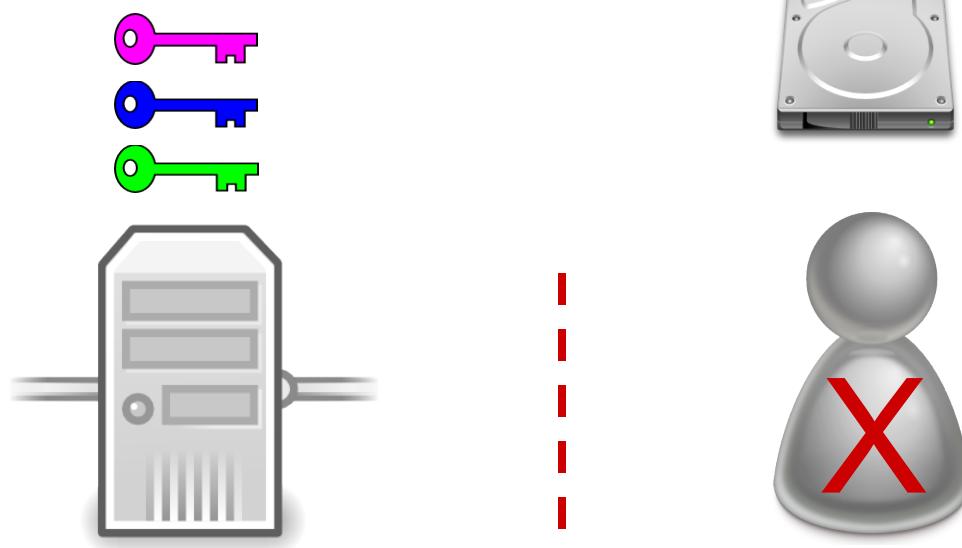
Example: Revoke Shared Access



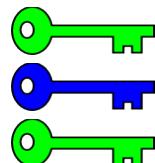
Example: Revoke Shared Access



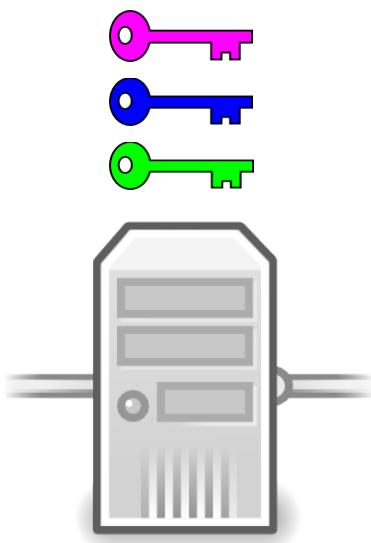
Example: Revoke Shared Access



140813: Bob Accessed
140906: Bob Accessed
141003: Bob Accessed



Example: Revoke Shared Access



140813: Bob Accessed



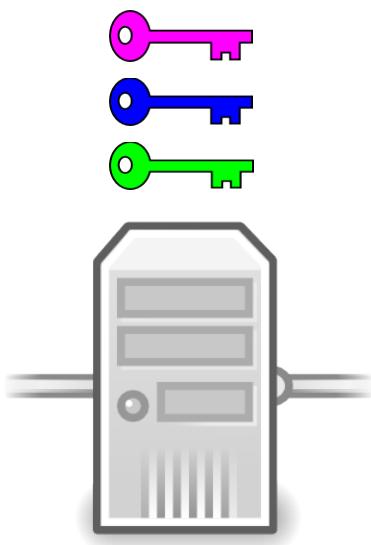
140906: Bob Accessed



141003: Bob Accessed



Example: Revoke Shared Access



140813: Bob Accessed



140906: Bob Accessed



141003: Bob Accessed

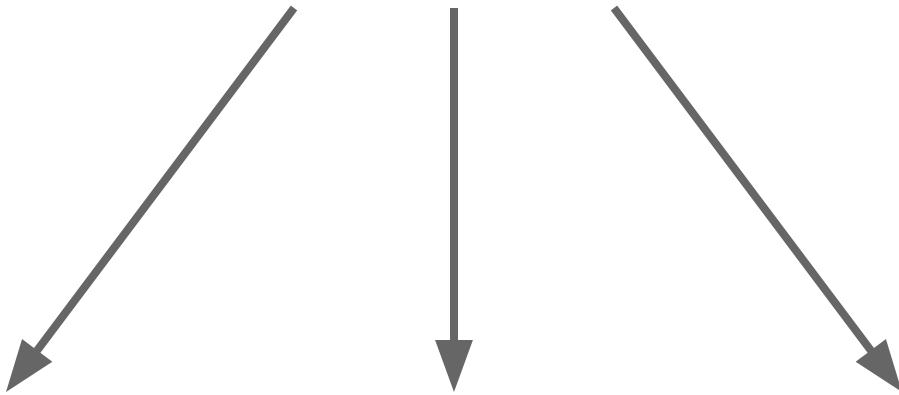


Auditing Benefits

140813: Bob Accessed Key A

140906: Bob Accessed Key B

141003: Bob Accessed Key A



Revocation
Semantics

Intrusion
Detection

Compliance
Verification

Autonomous Server Boot

Autonomous Server Boot



Autonomous Server Boot



Autonomous Server Boot



Untrusted

Autonomous Server Boot + Encryption



Autonomous Server Boot + Encryption



Autonomous Server Boot + Encryption



CorrectHorseBatteryStaple



Autonomous Server Boot + Encryption



CorrectHorseBatteryStaple



Autonomous Server Boot + Encryption



CorrectHorseBatteryStaple



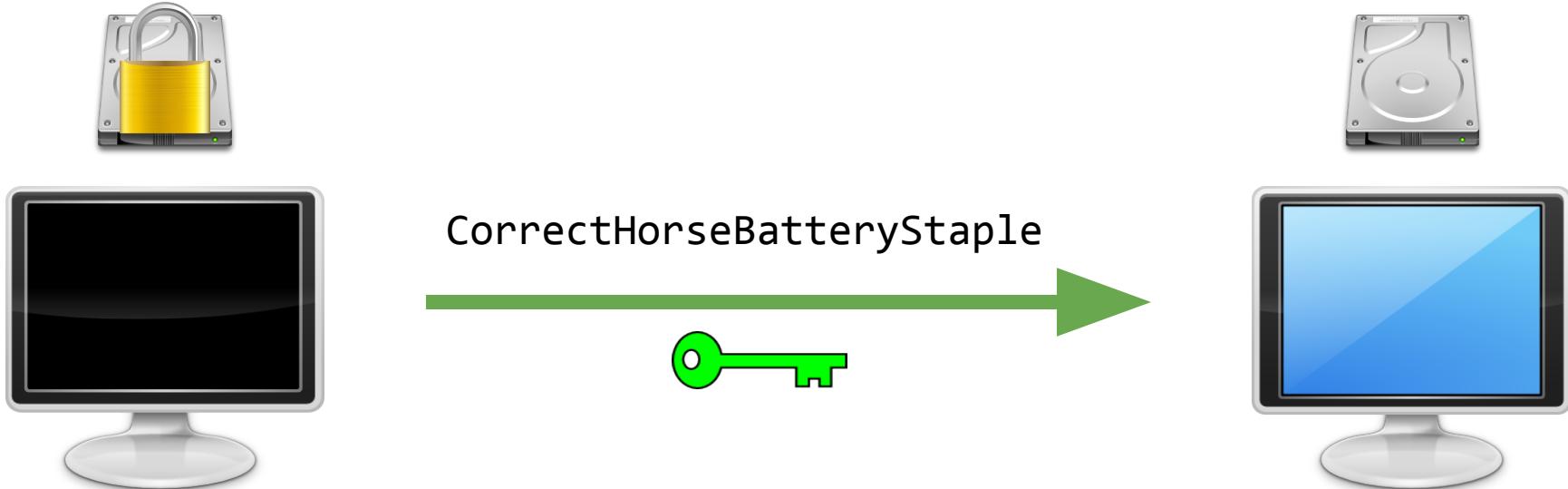
Autonomous Server Boot + Encryption



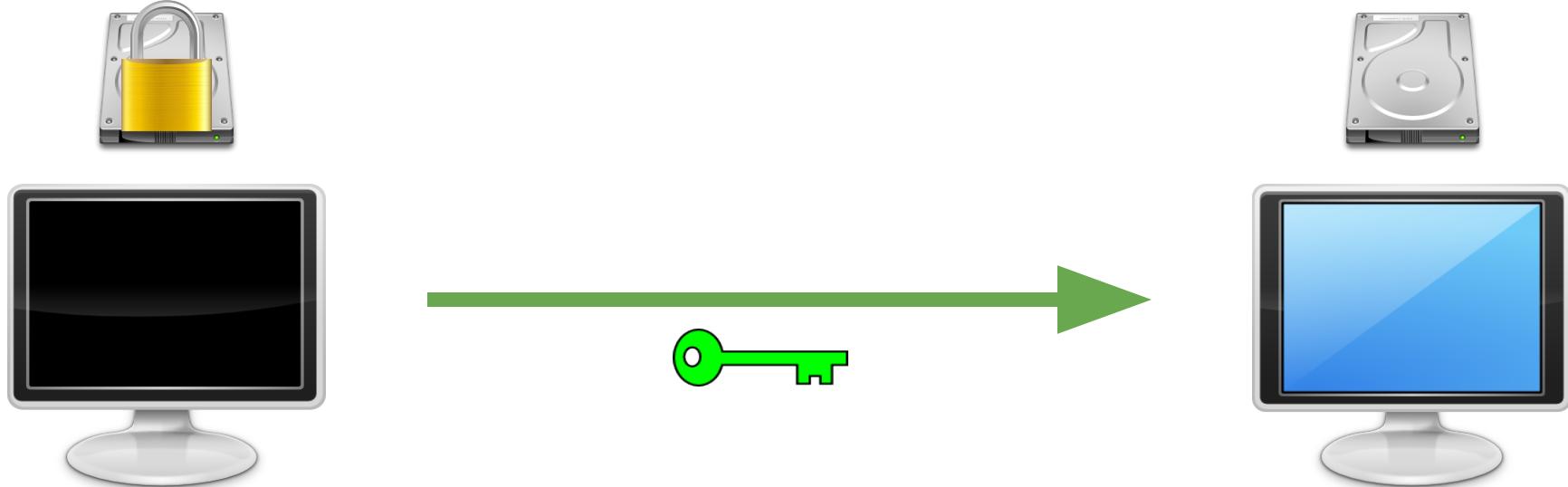
CorrectHorseBatteryStaple



Autonomous Server Boot + Encryption



Autonomous Server Boot + Encryption



Autonomous Server Boot + Encryption

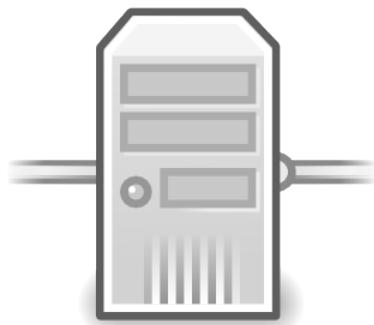


Autonomous Server Boot + Encryption



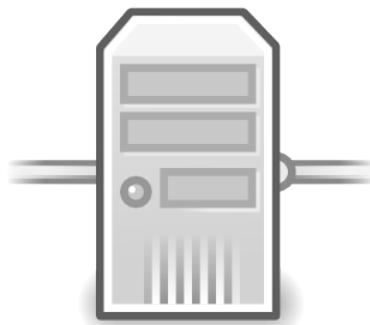
Autonomous Server Boot + Encryption + SSaaS





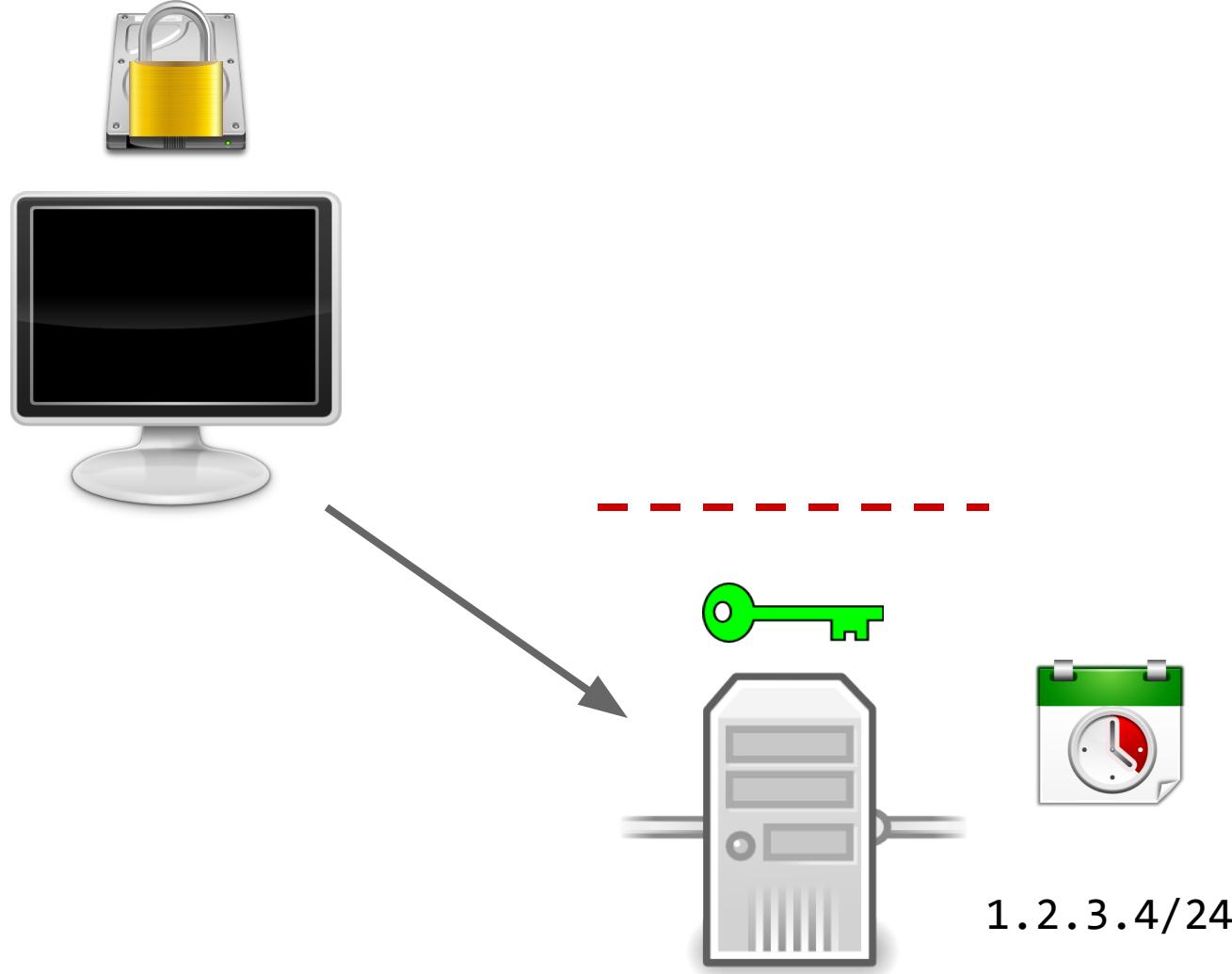
Autonomous Server Boot + Encryption + SSaaS



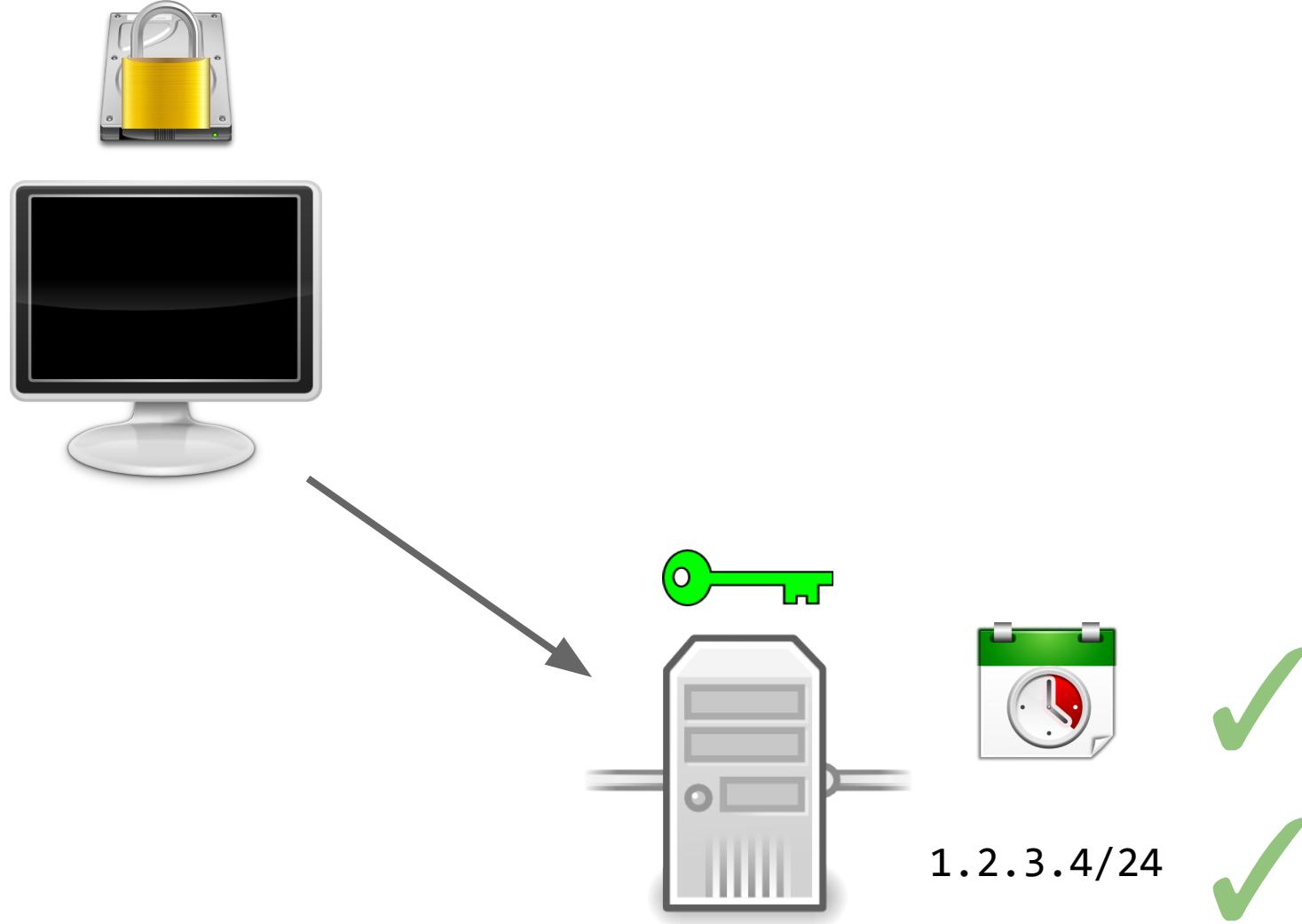


1.2.3.4/24

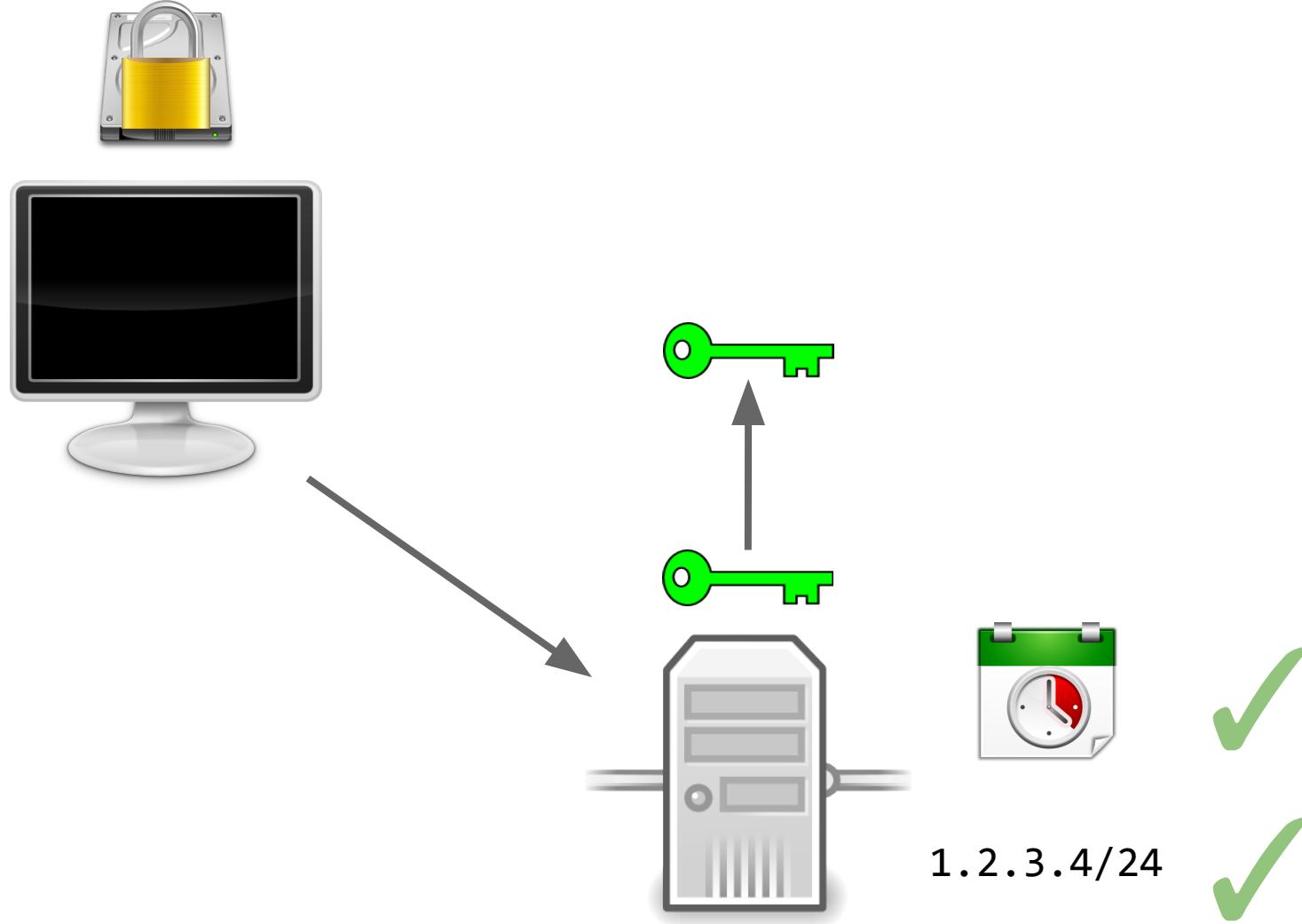
Autonomous Server Boot + Encryption + SSaaS



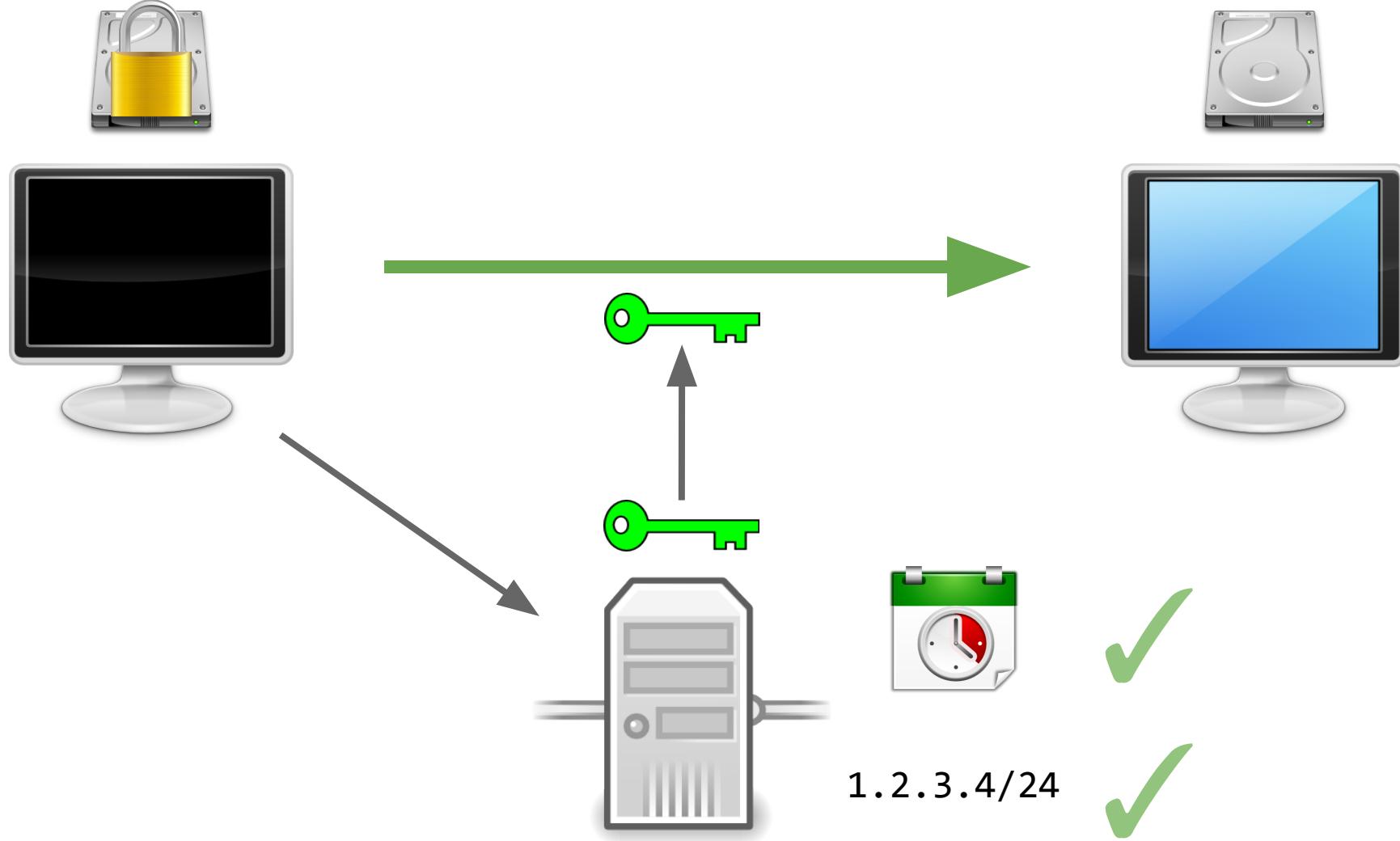
Autonomous Server Boot + Encryption + SSaaS



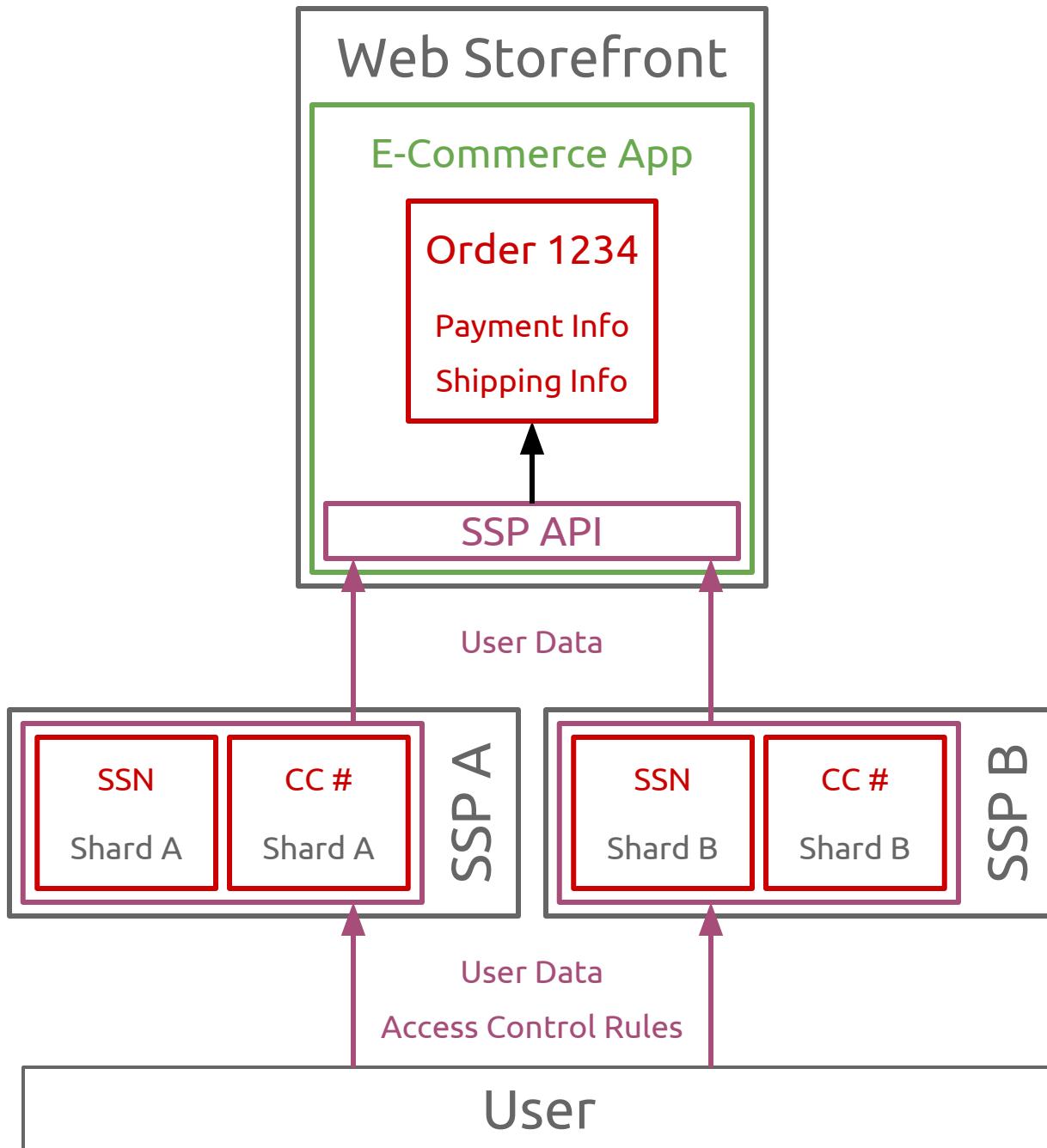
Autonomous Server Boot + Encryption + SaaS



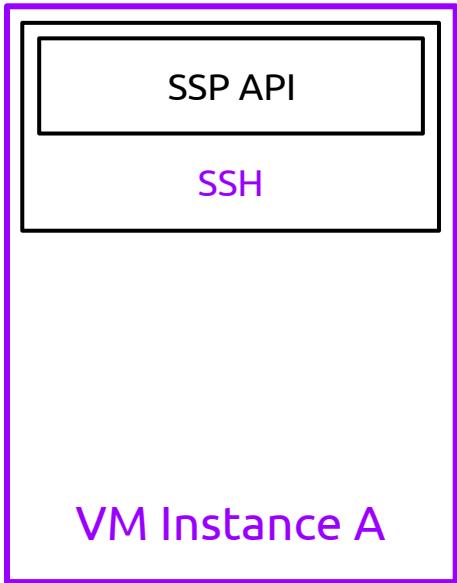
Autonomous Server Boot + Encryption + SaaS

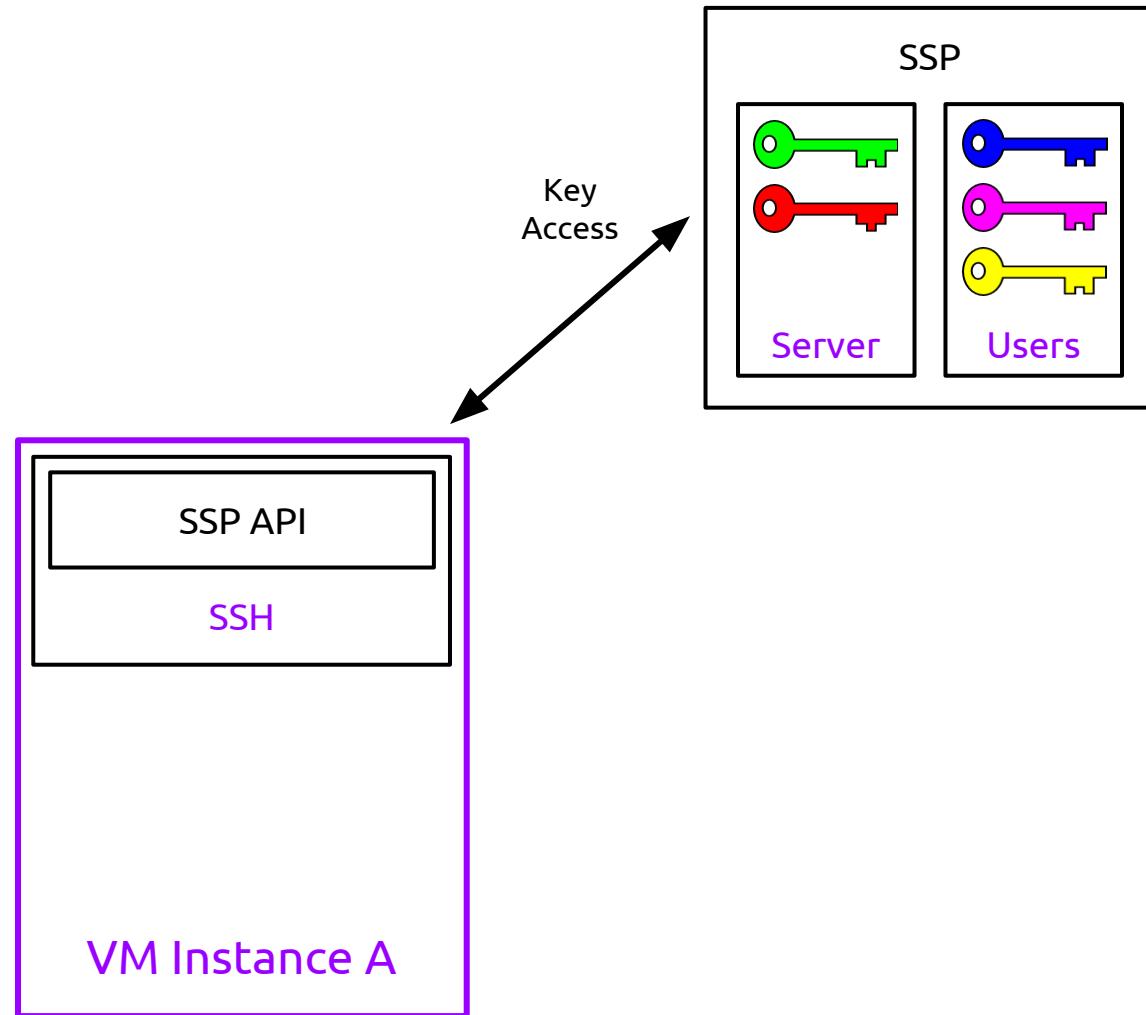


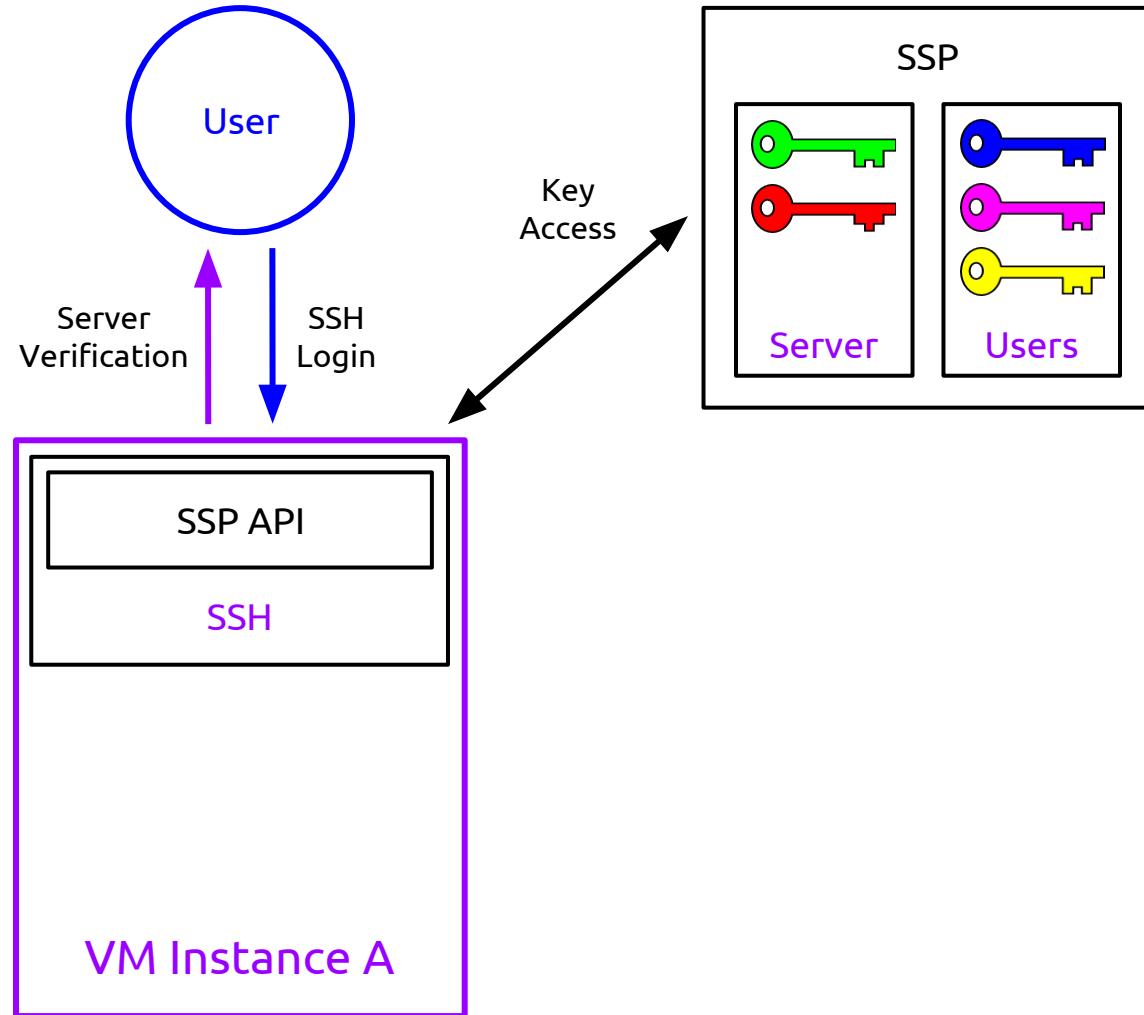
Personal Data Repository

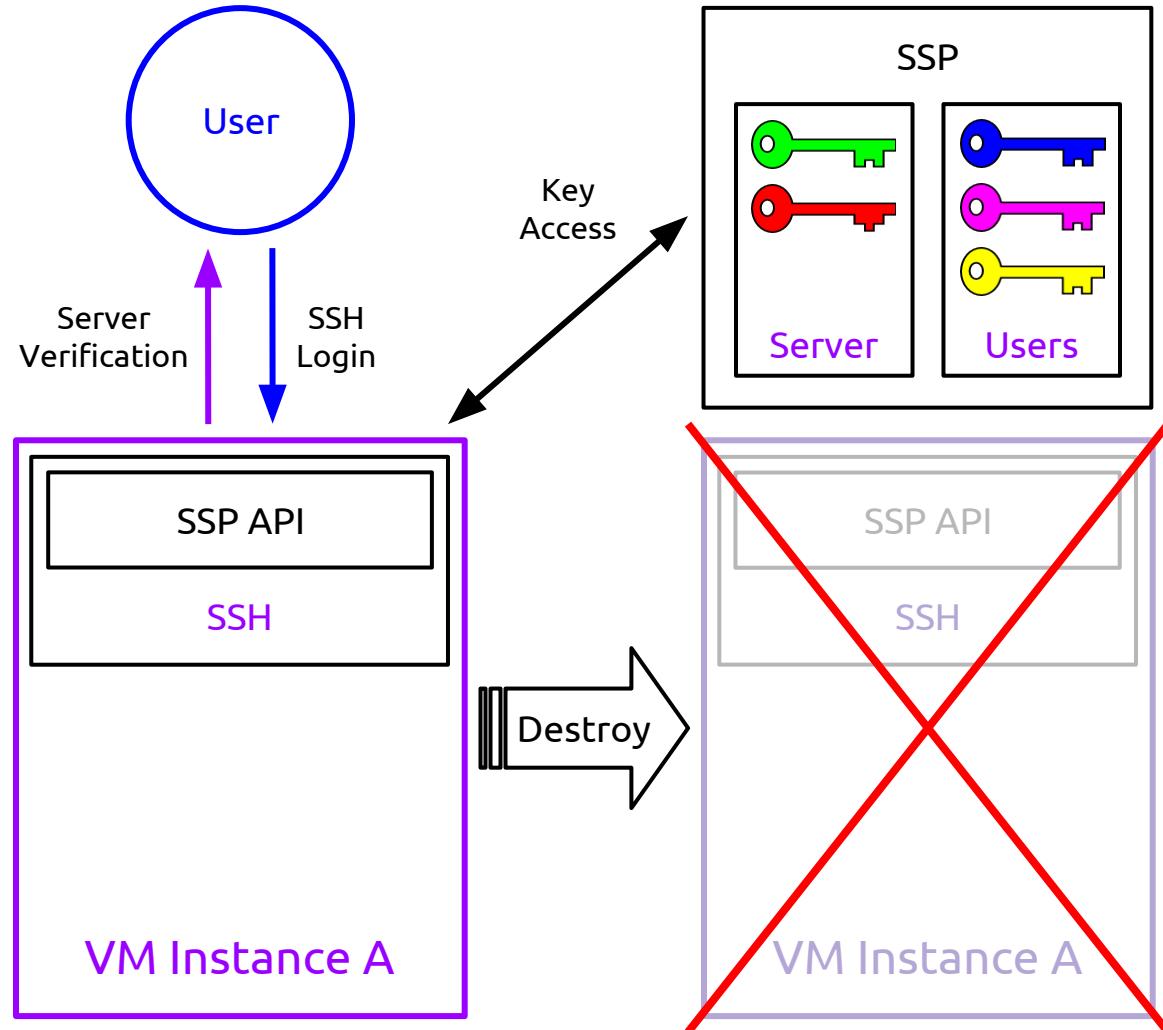


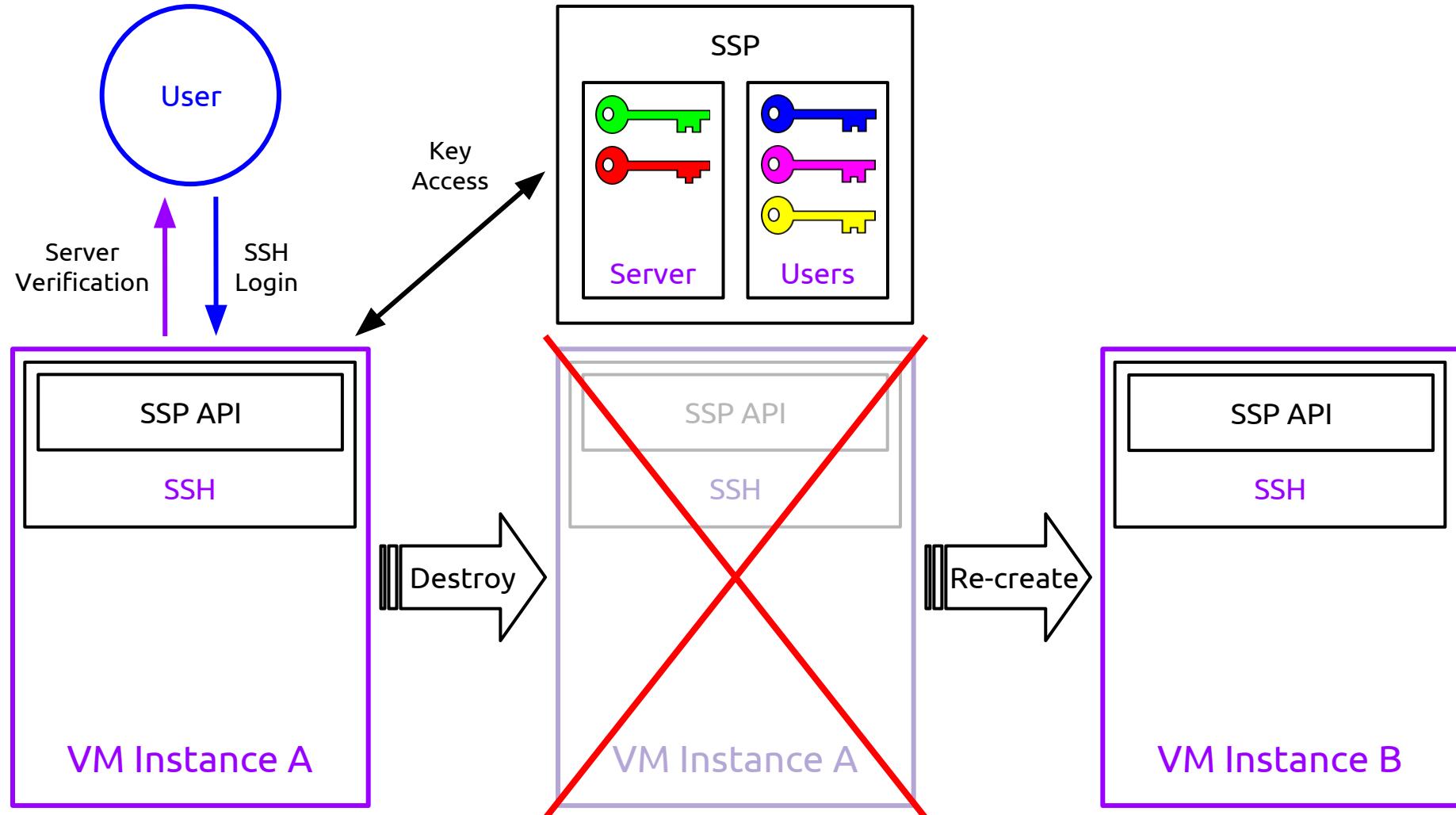
SSH Server Key Management

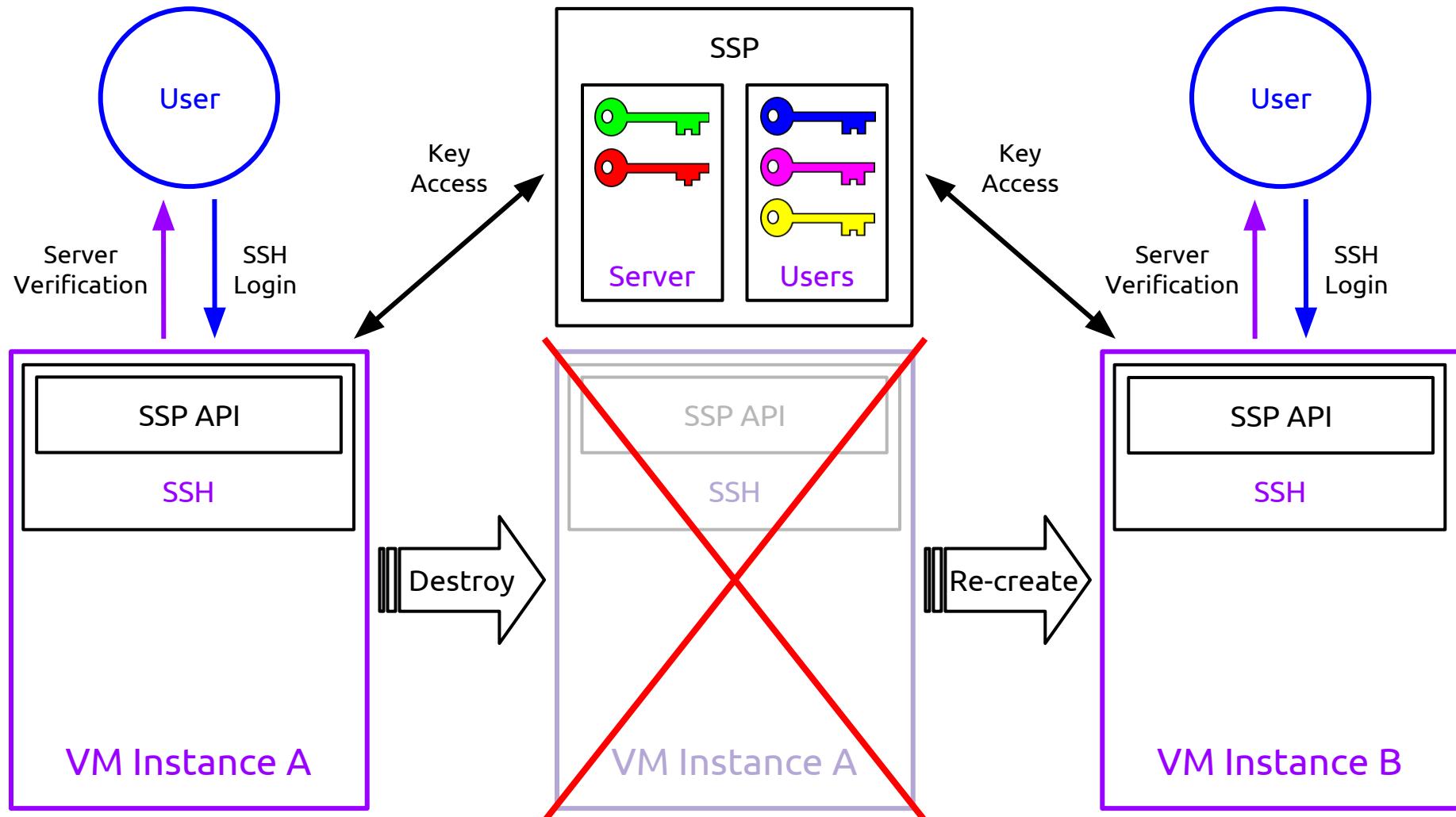












Management Server

