

# PCI DSS: Securing Data, Systems, and Applications

---

## REQUIREMENT 3: STORAGE OF CARDHOLDER DATA



**John Elliott**

PRIVACY, PAYMENTS, SECURITY AND RISK SPECIALIST

@withoutfire [www.withoutfire.com](http://www.withoutfire.com)



## Requirement 3



**Don't store cardholder data you don't need**

**Never store SAD after authorization**

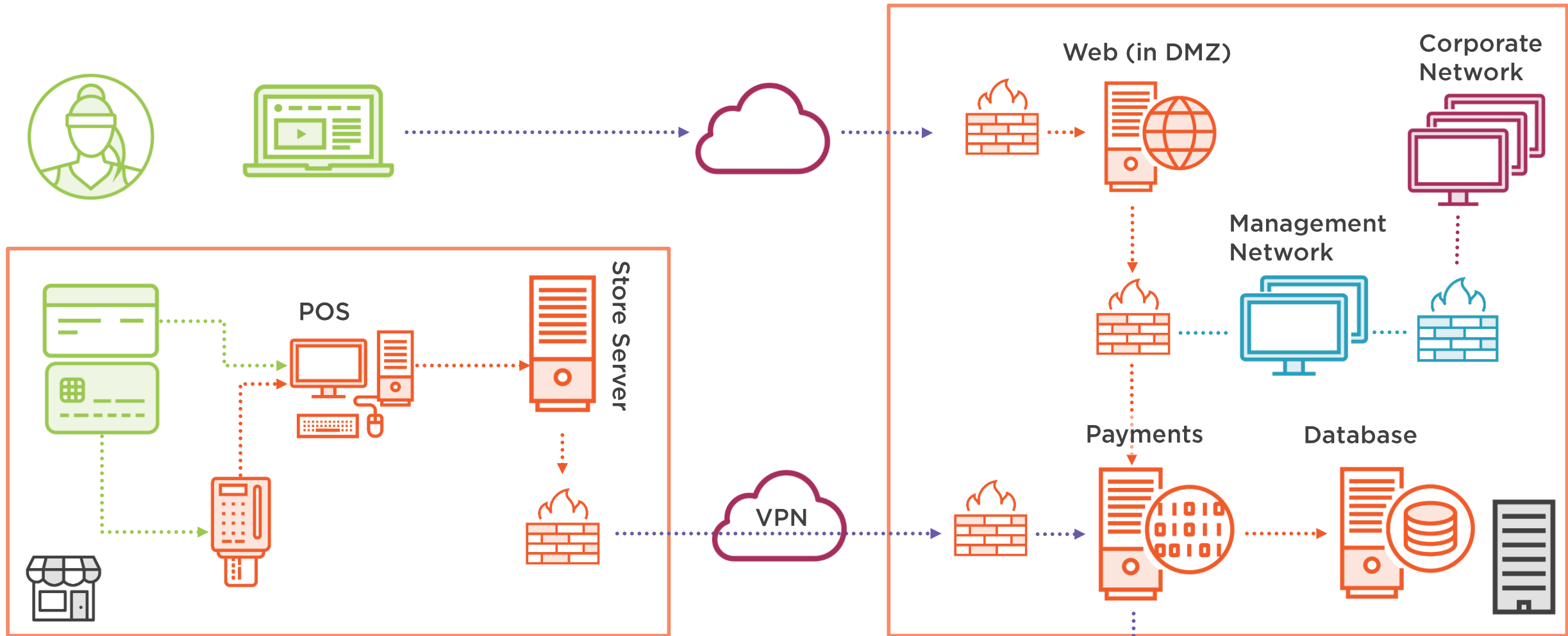
**Mask PAN when it is displayed**

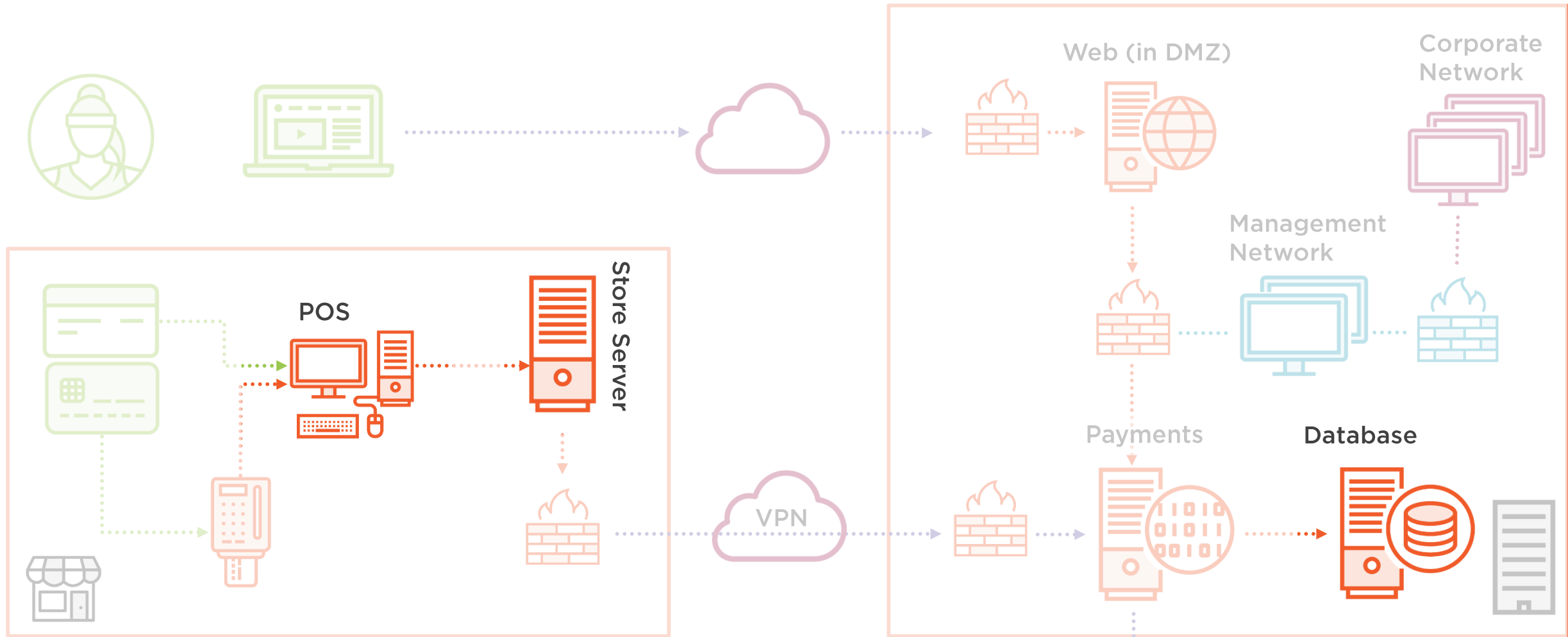
**Protect stored cardholder data**

**Do encryption properly**

**Have policies and procedures**







- Cardholder's card and systems
- Stores, processes or transmits
- Connected to or security affecting
- Out of scope of PCI DSS

- Unencrypted cardholder data
- Encrypted cardholder data
- Physical card read
- Management network
- Corporate network





## Requirement 3.1

Keep cardholder data storage  
to a minimum

If you don't need it, don't store it



### Requirement 3.1

Keep cardholder data storage to a minimum by implementing **data retention** and **disposal** policies, procedures and processes ...

1. Define **necessary retention**
2. Processes for **secure deletion**
3. **Quarterly** process for **deletion**



### Requirement Guidance

- A formal data retention policy identifies what data needs to be retained, and where that data resides so it can be securely destroyed or deleted as soon as it is no longer needed.
- The only CHD that may be stored after authorization is the primary account number or PAN (rendered unreadable), expiration date, cardholder name, and service code.
- Understanding where CHD is located is necessary so it can be properly retained or disposed of when no longer needed.

## 3.1 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Files</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Policies, procedures and processes</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Responsible people</b>





## Requirement 3.2

Do not store Sensitive Authentication Data (SAD) after authorization

And if you do, get rid of it after authorization





# Sensitive Authentication Data

		Data Element	Storage Permitted	Render Stored Data Unreadable per Requirement 3.4
Account Data	Cardholder Data	Primary Account Number (PAN)	Yes	Yes
		Cardholder Name	Yes	No
		Service Code	Yes	No
		Expiration Date	Yes	No
	Sensitive Authentication Data <sup>2</sup>	Full Track Data <sup>3</sup>	No	Cannot store per Requirement 3.2
		CAV2/CVC2/CVV2/CID <sup>4</sup>	No	Cannot store per Requirement 3.2
		PIN/PIN Block <sup>5</sup>	No	Cannot store per Requirement 3.2





### Requirement 3.2

**Do not store sensitive authentication data after authorization** (even if encrypted). If sensitive authentication data is received, render all data unrecoverable upon completion of the authorization process.



### Requirement Guidance

- Sensitive authentication data consists of full track data, card validation code or value, and PIN data. Storage of sensitive authentication data after authorization is prohibited!
- All PCI DSS requirements apply to issuers, and the only exception for issuers and issuer processors is that sensitive authentication data may be retained if there is a legitimate reason to do so.
- For non-issuing entities, retaining sensitive authentication data post-authorization is not permitted.



## 3.2 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Data stores &amp; system configurations</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Policies and procedures, any business justification</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Responsible people</b>



# Payment Card Security, Processing, and the PCI Standards

★★★★★ By John Elliott

Billions of card payment transactions happen every day. This course explains how they work, how criminals can turn payment card transactions into money, and how the PCI Security Standards that aim to stop them.

START A FREE 10-DAY TRIAL

▶ PLAY COURSE OVERVIEW

## Course Overview

Course Overview

1m

## Discovering How Card Payments Work

🔒 How Payment Card Transactions Appear on a Statement

4m

🔒 The Data Stored on a Payment Card

5m

🔒 The Entities Involved in Authorizing a Payment Card Transaction

5m

🔒 Getting Paid: Clearing and Settlement

3m

🔒 E-commerce Transactions and Payment Service Providers

5m





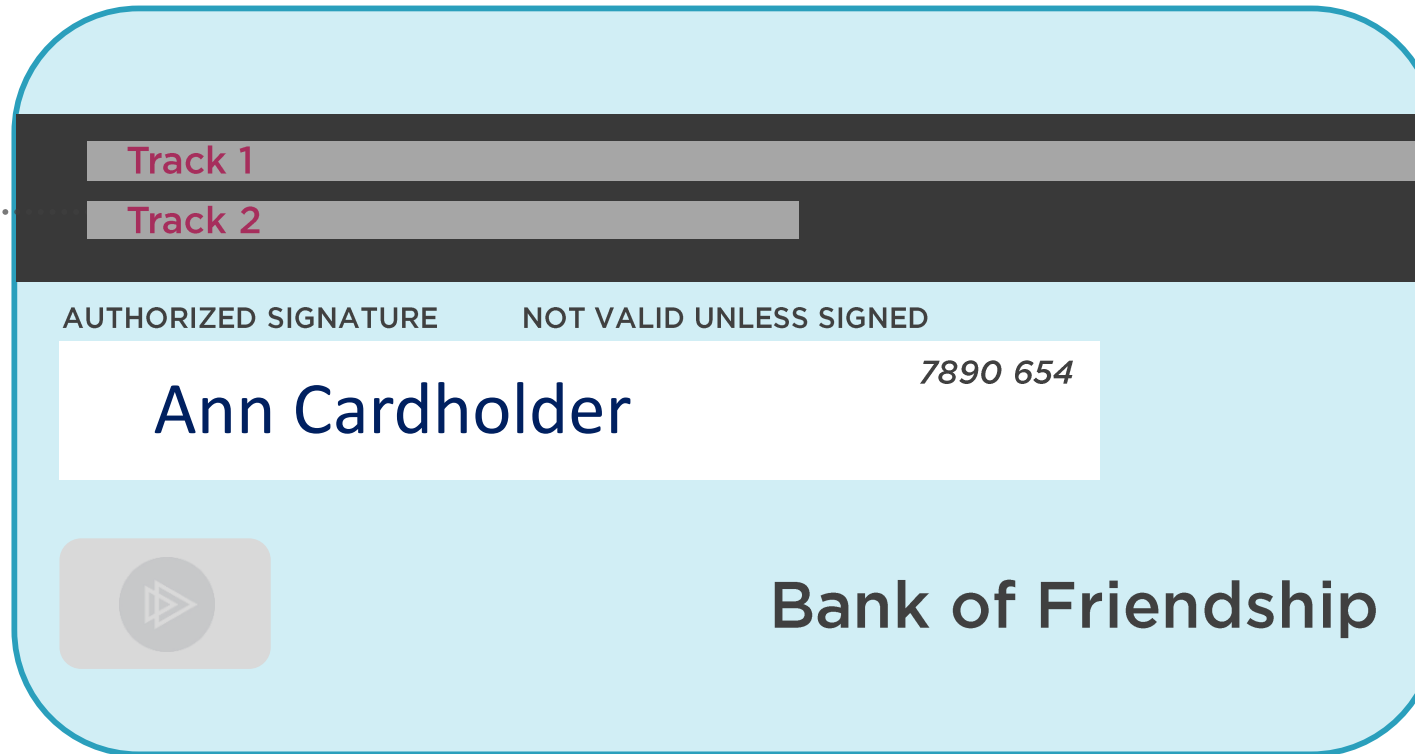
## Requirement 3.2.1

Do not store all the track or chip data



# The Data on a Payment Card

Magnetic  
stripe  
(track data)



# The Data on a Payment Card





### Requirement 3.2.1

**Do not store** the **full** contents of any **track** (from the magnetic stripe located on the back of a card, equivalent data contained on a chip, or elsewhere) **after authorization.**

This data is alternatively called full track, track, track 1, track 2, and magnetic-stripe data.



### Requirement Guidance

- If full track data is stored, malicious individuals who obtain that data can use it to reproduce payment cards and complete fraudulent transactions.





## 3.2.1 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Incoming transaction data, database contents, log/trace files</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Database schema</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>-</b>	



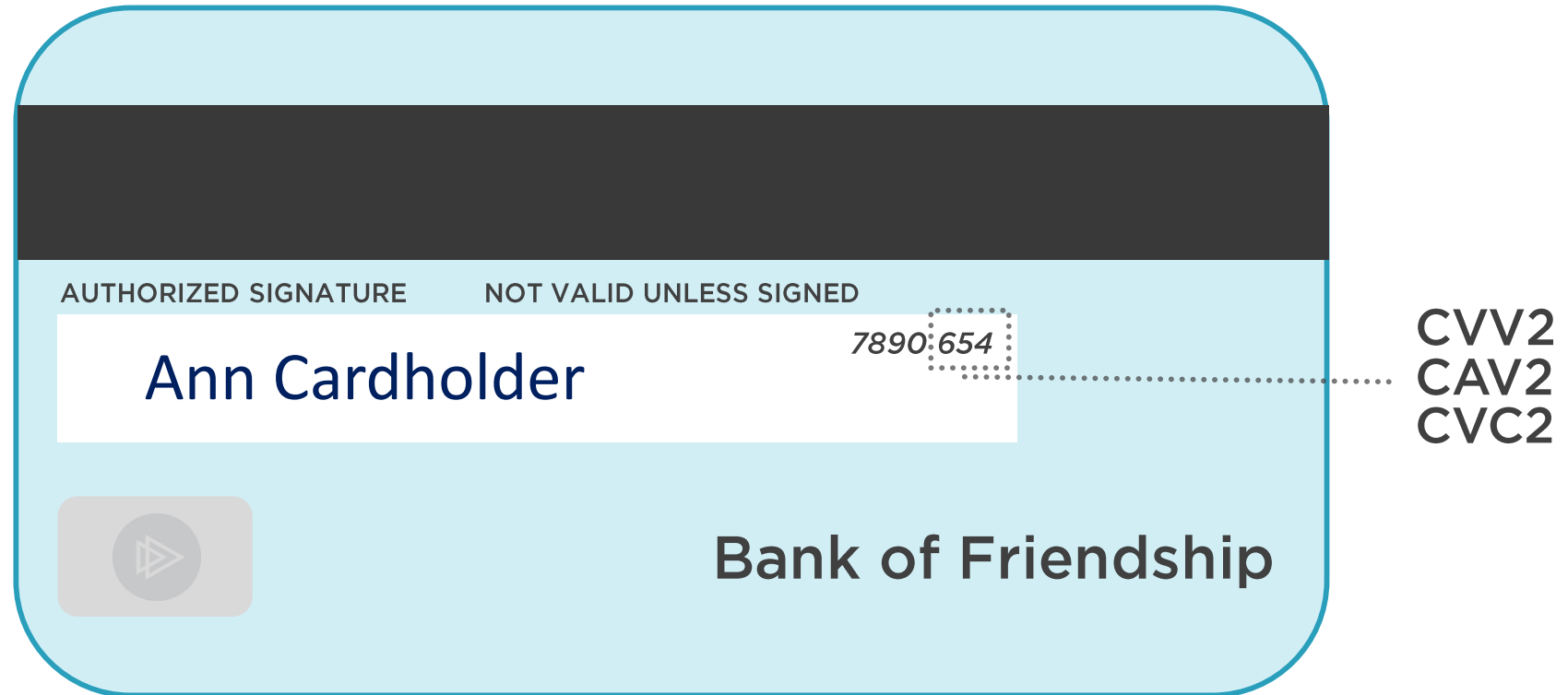


## Requirement 3.2.2

Do not store the card verification code  
after authentication



# The Data on a Payment Card



# The Data on a Payment Card





### Requirement 3.2.2

**Do not store the card verification code or value after authorization.**

(three-digit or four-digit number printed on the front or back of a payment card used to verify card-not-present transactions)



### Requirement Guidance

- The purpose of the card validation code is to protect "card-not-present" transactions—Internet or mail order/telephone order (MO/TO) transactions—where the consumer and the card are not present.
- If this data is stolen, malicious individuals can execute fraudulent Internet and MO/TO transactions.



## 3.2.2 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Incoming transaction data, database contents, log/trace files</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Database schema</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>-</b>	





## Requirement 3.2.3

Do not store the PIN after authentication



### Requirement 3.2.3

**Do not store** the personal identification number (**PIN**) or the encrypted PIN block **after authorization**.



### Requirement Guidance

- These values should be known only to the card owner or bank that issued the card. If this data is stolen, malicious individuals can execute fraudulent PIN-based debit transactions (for example, ATM withdrawals).





## 3.2.3 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Incoming transaction data, database contents, log/trace files</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Database schema</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>-</b>	





## Requirement 3.3

Mask PAN when displayed

Only those who must see more  
for their role should see more





### Requirement 3.3

**Mask PAN when displayed** (the first six and last four digits are the maximum number of digits to be displayed), such that **only** personnel with a **legitimate business need** can **see more** than the first six/last four digits of the PAN.

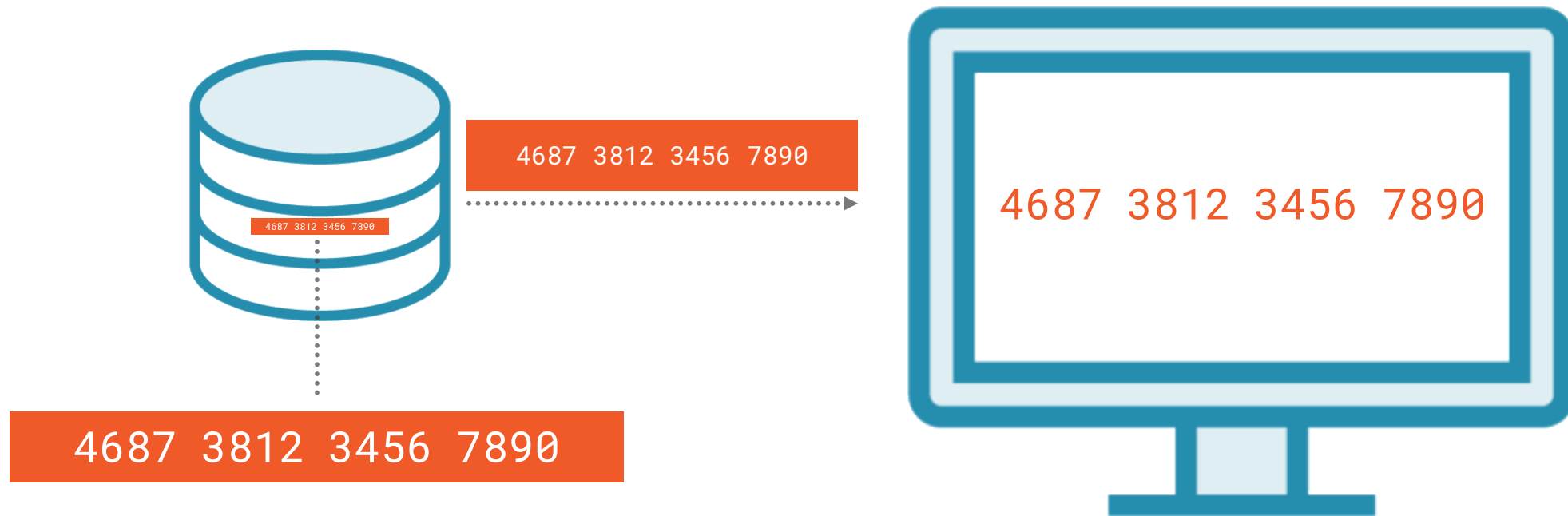


### Requirement Guidance

- Ensuring that full PAN is only displayed for those with a legitimate business need to see the full PAN minimizes the risk of unauthorized persons gaining access to PAN data.
- The masking approach should always ensure that only the minimum number of digits is displayed as necessary to perform a specific business function.



# Understanding Masking



Not masked. System now processes PAN and so is in scope of PCI DSS



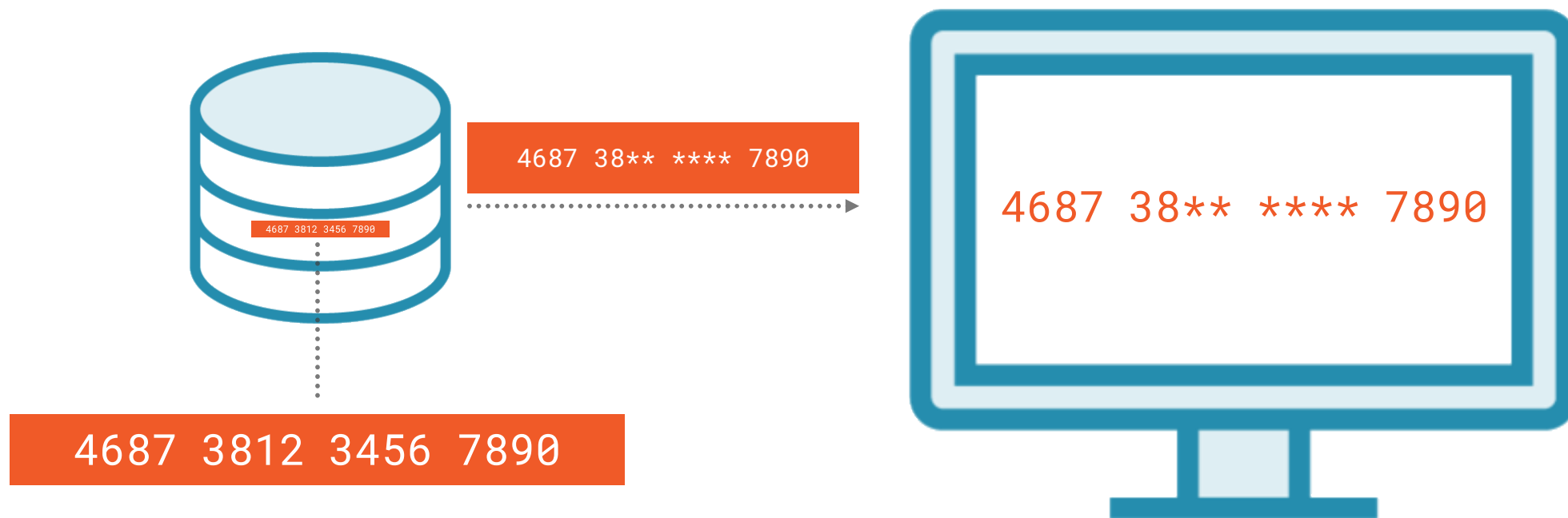
# Understanding Masking



Masked. System does not processes PAN and so NOT in scope of PCI DSS



# Understanding Masking



Still masked. System does not processes PAN and so NOT in scope of PCI DSS



# Three Common Masking Myths



Merchant receipts



Customers shopping  
online



Contact / call centers

**“legitimate business need”**



## 3.3 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>System configurations, documents, displays</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Policies and procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>-</b>	







## Requirement 3.4

Don't store plaintext PAN





Please, just don't store  
cardholder data.





## Requirement 3.4

Don't store plaintext PAN





### Requirement 3.4

**Render PAN unreadable** anywhere it is stored (including on portable digital media, backup media, and in logs).

1. Hashing
2. Truncation
3. Tokens
4. Strong cryptography (3.5, 3.6)



### Requirement Guidance

- PANs stored in primary storage (databases, or flat files such as text files spreadsheets) as well as non-primary storage (backup, audit logs, exception or troubleshooting logs) must all be protected.



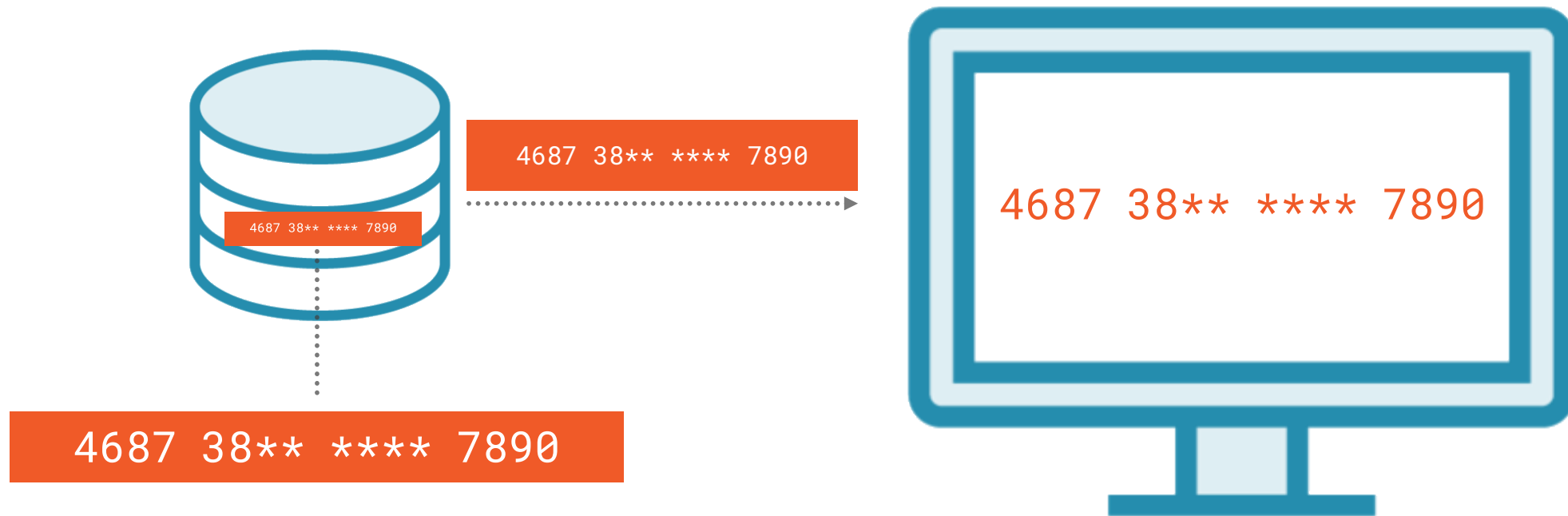
# Hashing

**Must hash the whole PAN**

**Must use recognized  
(e.g. NIST) algorithm**



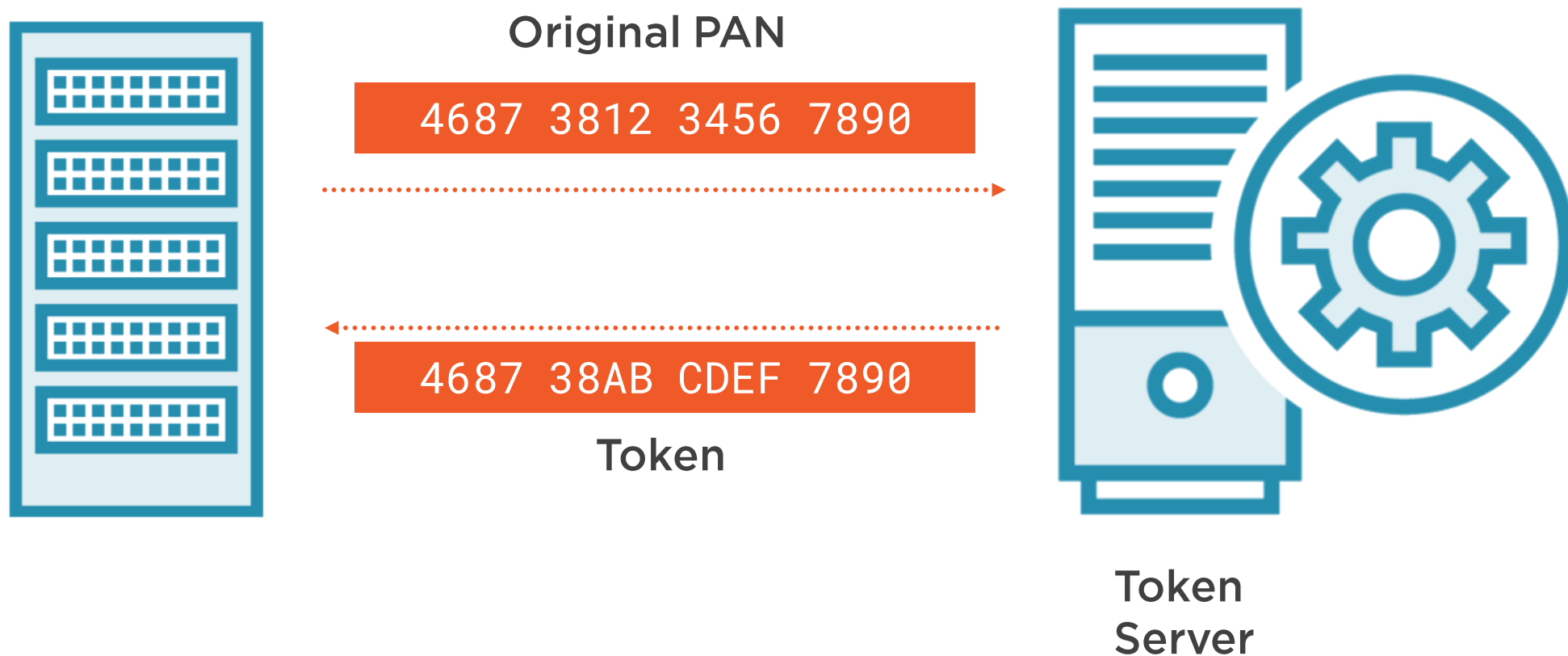
# Understanding Truncation



First 6, last 4 see FAQ 1091



# Understanding Tokenization





Done properly hashing, truncation and tokenization are magic

PAN = Cardholder data

Hashed PAN **NOT** cardholder data

Truncated PAN **NOT** cardholder data

Tokenized PAN **NOT** cardholder data







Encryption is not magic

PAN = Cardholder data

Encrypted PAN = cardholder data



“Cryptography based on industry-tested and accepted algorithms, along with key lengths that provide a minimum of 112-bits of effective key strength and proper key-management practices”

## **Strong Cryptography**

[www.pcisecuritystandards.org/documents/PCI\\_DSS\\_Glossary\\_v3-2.pdf](http://www.pcisecuritystandards.org/documents/PCI_DSS_Glossary_v3-2.pdf)

*See: NIST Special Publication 800-57 Part 1*



## 3.4 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Data repository files and/or tables, removable media, logs and audit logs</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Relevant incl. vendors</b>
<b>Examine records</b>	<b>Y</b>	<b>-</b>
<b>Interview people</b>	<b>-</b>	





## Requirement 3.4.1

If you're relying on whole disk encryption,  
manage disk encryption  
independently from O/S authentication  
and access control mechanisms





### Requirement 3.4.1

If **disk encryption** is used (rather than file- or column-level database encryption), **logical access** must be managed **separately** and **independently** of native operating system authentication and access control mechanisms (for example, by not using local user account databases or general network login credentials). Decryption keys must not be associated with user accounts.



### Requirement Guidance

- To be compliant with this requirement, disk-level encryption cannot:
- 1) Use the same user account authenticator as the operating system, or
- 2) Use a decryption key that is associated with or derived from the system's local user account database or general network login credentials.
- Full disk encryption helps to protect data in the event of physical loss of a disk and therefore may be appropriate for portable devices that store cardholder data.



## 3.4.1 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Configuration and authentication process, including on removable media</b>
<b>Examine documentation</b>	<b>-</b>	
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Responsible people</b>





## Requirement 3.5

Look after the keys ...



**If you do not store  
cardholder data, you  
will not need to do this**



Get a Hardware  
Security Module (HSM):  
Anything else is optimism







### Requirement 3.5

Document and implement procedures to **protect keys** used to secure stored cardholder data **against disclosure and misuse.**



### Requirement Guidance

- Cryptographic keys must be strongly protected because those who obtain access will be able to decrypt data. Key-encrypting keys, if used, must be at least as strong as the data-encrypting key in order to ensure proper protection of the key that encrypts the data as well as the data encrypted with that key.
- The requirement to protect keys from disclosure and misuse applies to both data-encrypting keys and key-encrypting keys. Because one key-encrypting key may grant access to many data-encrypting keys, the key-encrypting keys require strong protection measures.



## 3.5 Testing Procedures

<b>Observe/examine systems and settings</b>	-	
<b>Examine documentation</b>	Y	<b>Key-management policies and procedures</b>
<b>Examine records</b>	-	
<b>Interview people</b>	-	





## Requirement 3.5.1

**Service providers** must also maintain a documented description of the cryptographic architecture





### Requirement 3.5.1

#### *Service providers*

#### ***additional requirement:***

Maintain a documented description of the cryptographic architecture that includes:

Details of all algorithms, protocols, and keys used to protect CHD, including key strength and expiry date

Description of the key

usage for each key

Inventory of HSMs + other

SCDs for key management.



### Requirement Guidance

- Maintaining current documentation of the cryptographic architecture enables an entity to understand the algorithms, protocols, and cryptographic keys used to protect cardholder data, as well as the devices that generate, use and protect the keys. This allows an entity to keep pace with evolving threats to their architecture, enabling them to plan for updates as the assurance levels provided by different algorithms/key strengths changes. Maintaining such documentation also allows an entity to detect lost or missing keys or key-management devices, and identify unauthorized additions to their cryptographic architecture.



## 3.5.1 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>-</b>	
<b>Examine documentation</b>	<b>Y</b>	<b>Cryptographic architecture document</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Responsible people</b>





## Requirement 3.5.2

The fewer people with access to keys, the better





### Requirement 3.5.2

Restrict access to cryptographic keys to the **fewest** number of **custodians** necessary.



### Requirement Guidance

- There should be very few who have access to cryptographic keys (reducing the potential for rendering cardholder data visible by unauthorized parties), usually only those who have key custodian responsibilities.

## 3.5.2 Testing Procedures

<b>Observe/examine systems and settings</b>	-	
<b>Examine documentation</b>	-	
<b>Examine records</b>	Y	User access lists
<b>Interview people</b>	-	







## Requirement 3.5.3

Protect the data encryption  
/ decryption keys





### Requirement 3.5.3

Store secret and private keys used to encrypt / decrypt cardholder data in one (or more) of these ways.

1. Use a **key-encrypting key** (KEK)
2. Use an **HSM**
3. Use **two** or more **full length** key **components**



### Requirement Guidance

- Cryptographic keys must be stored securely to prevent unauthorized or unnecessary access that could result in the exposure of cardholder data.
- It is not intended that the key-encrypting keys be encrypted, however they are to be protected against disclosure and misuse as defined in Requirement 3.5. If key-encrypting keys are used, storing the key-encrypting keys in physically and/or logically separate locations from the data- encrypting keys reduces the risk of unauthorized access to both keys.



## 3.5.3 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>System configurations, key storage locations</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Cryptographic key procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>-</b>	



## Courses

**Cryptography Fundamentals for Developers and Security Professionals**

by Michael Perry

Intermediate

May 15 2014

4h 14m

★★★★☆ (281)

**Practical Cryptography in .NET**

by Stephen Haunts

Intermediate

May 20 2015

3h 59m

★★★★☆ (235)

**Cryptography: The Big Picture**

by Matt Glass

Intermediate

Jun 6 2017

1h 24m

★★★★☆ (88)

**Introduction to Cryptography in .NET**

by Robert Boedighe...

Intermediate

Apr 29 2013

2h 3m

★★★★☆ (436)

**Ethical Hacking: Cryptography**

by Dale Meredith

Beginner

Dec 2 2018

2h 28m

★★★★☆ (12)

**Practical Cryptography in Node.js**

by Justin Boyer

Intermediate

Feb 4 2019

1h 22m

★★★★☆ (21)





## Requirement 3.5.4

Don't leave multiple copies of keys lying around your systems





#### Requirement 3.5.4

Store cryptographic keys in the **fewest possible locations**.



#### Requirement Guidance

- Storing cryptographic keys in the fewest locations helps an organization to keep track and monitor all key locations, and minimizes the potential for keys to be exposed to unauthorized parties.



## 3.5.4 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Key storage locations</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Key management processes</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>-</b>	





# Requirement 3.6

Document and manage your keys ...







If you have not done this  
before, don't try to make it up!

Ask a specialist.





## Requirement 3.6

Fully **document** and **implement** all key-management **processes and procedures** for cryptographic keys used for encryption of cardholder data, including the following:

*[Items 3.6.1 through 3.6.8]*



## Requirement Guidance

- The manner in which cryptographic keys are managed is a critical part of the continued security of the encryption solution. A good key-management process, whether it is manual or automated as part of the encryption product, is based on industry standards and addresses all key elements at 3.6.1 through 3.6.8.
- Providing guidance to customers on how to securely transmit, store and update cryptographic keys can help prevent keys from being mismanaged or disclosed to unauthorized entities.

## 3.6 Testing Procedures

<b>Observe/examine systems and settings</b>	-	
<b>Examine documentation</b>	Y	Processes and procedures
<b>Examine records</b>	-	
<b>Interview people</b>	-	





## Requirement 3.6.1

Generation of strong cryptographic keys





### Requirement 3.6.1

*(Document and implement)*  
**Generation** of **strong**  
cryptographic keys.



### Requirement Guidance

- The encryption solution must generate strong keys, **as defined in the *PCI DSS and PA-DSS Glossary of Terms, Abbreviations, and Acronyms*** under "Cryptographic Key Generation."
- Use of strong cryptographic keys significantly increases the level of security of encrypted cardholder data.

## 3.6.1 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Key-generation procedure</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Processes and procedures</b>
<b>Examine records</b>	<b>!</b>	<b>Documentary evidence produced when you generated keys</b>
<b>Interview people</b>	<b>-</b>	





## Requirement 3.6.2

Secure cryptographic key distribution





### Requirement 3.6.2

*(Document and implement)*  
**Secure** cryptographic key  
**distribution.**



### Requirement Guidance

- The encryption solution must distribute keys securely, meaning the keys are distributed only to custodians identified in Requirement 3.5.2, and are never distributed in the clear.



## 3.6.2 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Key distribution!</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Distribution procedures</b>
<b>Examine records</b>	<b>!</b>	<b>Of key distribution</b>
<b>Interview people</b>	<b>!</b>	<b>Key custodians</b>





## Requirement 3.6.3

Secure cryptographic key storage





### Requirement 3.6.3

*(Document and implement)*  
**Secure** cryptographic key  
**storage.**



### Requirement Guidance

- The encryption solution must store keys securely, for example, by encrypting them with a key-encrypting key. Storing keys without proper protection could provide access to attackers, resulting in the decryption and exposure of cardholder data.



## 3.6.3 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Key stores</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Key storage procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>-</b>	





## Requirement 3.6.4

Cryptographic key changes for keys that have reached the end of their crypto-period

# Crypto-period



Which encryption algorithm is used

Size of the key

Cryptanalysis – how successful people have been at breaking encryption

What data has been encrypted with a single key



#### Requirement 3.6.4

*(Document and implement)* Cryptographic **key changes** for keys that have reached the **end** of their **cryptoperiod** (for example, after a defined period of time has passed and/or after a certain amount of cipher-text has been produced by a given key), as defined by the associated application vendor or key owner, and based on industry best practices and guidelines.



#### Requirement Guidance

- A cryptoperiod is the time span during which a particular cryptographic key can be used for its defined purpose. Considerations for defining the cryptoperiod include, but are not limited to, the strength of the underlying algorithm, size or length of the key, risk of key compromise, and the sensitivity of the data being encrypted.
- Periodic changing of encryption keys when the keys have reached the end of their cryptoperiod is imperative to minimize the risk of someone's obtaining the encryption keys, and using them to decrypt data.



## 3.6.4 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>-</b>	
<b>Examine documentation</b>	<b>Y</b>	<b>Key management procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Interview relevant people</b>







## Requirement 3.6.5

Retirement or replacement of “weakened” keys





### Requirement 3.6.5

*(Document and implement)*  
**Retirement** or **replacement**  
(for example, archiving,  
destruction, and/or  
revocation) of **keys** as  
deemed **necessary** when  
the **integrity** of the key has  
been weakened (for  
example, departure of an  
employee with knowledge  
of a clear-text key  
component), or keys are  
**suspected** of being  
**compromised**.



### Requirement Guidance

- Keys that are no longer used or needed, or keys that are known or suspected to be compromised, should be revoked and/or destroyed to ensure that the keys can no longer be used. If such keys need to be kept (for example, to support archived, encrypted data) they should be strongly protected.
- The encryption solution should provide for and facilitate a process to replace keys that are due for replacement or that are known to be, or suspected of being, compromised.



## 3.6.5 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>-</b>	
<b>Examine documentation</b>	<b>Y</b>	<b>Key-management procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Interview relevant people</b>





## Requirement 3.6.6

Split knowledge and dual control  
if there are clear-text keys



There should never  
be clear text keys.





### Requirement 3.6.6

*(Document and implement)*

If manual **clear-text** cryptographic key-management operations are used, these operations must be managed using **split knowledge** and **dual control**.

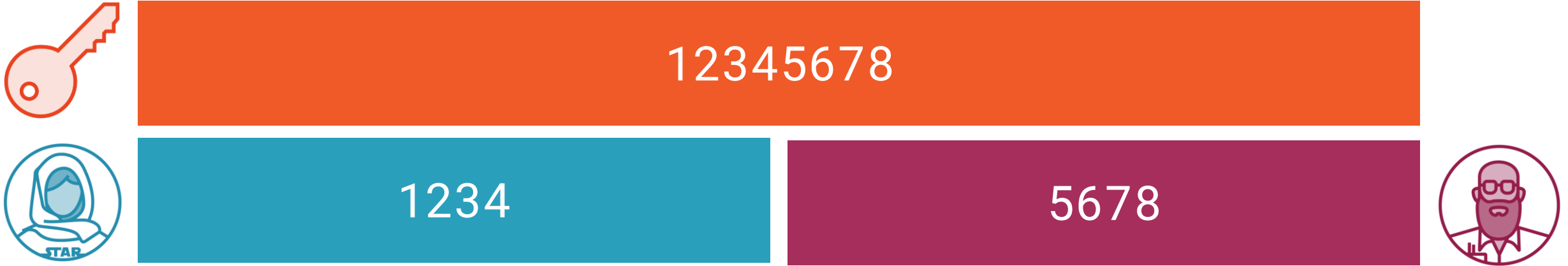


### Requirement Guidance

- **Split knowledge** is a method in which **two or more people** separately have **key components**, where each **person knows** only their **own key component**, and the individual key components convey no knowledge of the original cryptographic key.
- **Dual control** requires **two or more people to perform a function**, and no single person can access or use the authentication materials of another.



# Not Split Knowledge



# Split Knowledge



42536475

xor



37056341

=



12345678







### Requirement 3.6.6

*(Document and implement)*

If manual **clear-text** cryptographic key-management operations are used, these operations must be managed using **split knowledge** and **dual control**.



### Requirement Guidance

- **Split knowledge** is a method in which **two or more people** separately have **key components**, where each **person knows** only their **own key component**, and the individual key components convey no knowledge of the original cryptographic key.
- **Dual control** requires **two or more people to perform a function**, and no single person can access or use the authentication materials of another.



## 3.6.6 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>Y</b>	<b>Key management functions</b>
<b>Examine documentation</b>	<b>Y</b>	<b>Key-management procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Responsible people</b>





## Requirement 3.6.7

Prevent substitution of cryptographic keys





### Requirement 3.6.7

*(Document and implement)*  
**Prevention** of  
**unauthorized substitution**  
of cryptographic keys.



### Requirement Guidance

- The encryption solution should not allow for or accept substitution of keys coming from unauthorized sources or unexpected processes.

## 3.6.7 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>-</b>	
<b>Examine documentation</b>	<b>Y</b>	<b>Key-management procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Responsible people</b>





## Requirement 3.6.8

Cryptographic key custodian responsibilities





### Requirement 3.6.8

*(Document and implement)*  
Requirement for  
cryptographic **key**  
**custodians** to **formally**  
**acknowledge** that they  
understand and accept  
their key- custodian  
responsibilities.



### Requirement Guidance

- This process will help ensure individuals that act as key custodians commit to the key-custodian role and understand and accept the responsibilities.



## 3.6.8 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>-</b>	
<b>Examine documentation</b>	<b>Y</b>	<b>Key-management procedures, documents</b>
<b>Examine records</b>	<b>Y</b>	<b>Evidence of acknowledgement</b>
<b>Interview people</b>	<b>!</b>	<b>Amazingly not – but QSAs may ask...</b>







## Requirement 3.7

Have and use policies  
and operational procedures





### Requirement 3.7

Ensure that security policies and operational procedures for protecting stored cardholder data are **documented, in use**, and **known** to all affected parties.



### Requirement Guidance

- Personnel need to be aware of and following security policies and documented operational procedures for managing the secure storage of cardholder data on a continuous basis.



## 3.7 Testing Procedures

<b>Observe/examine systems and settings</b>	<b>-</b>	
<b>Examine documentation</b>	<b>Y</b>	<b>Security policies and operational procedures</b>
<b>Examine records</b>	<b>-</b>	
<b>Interview people</b>	<b>Y</b>	<b>Responsible people</b>



# That's Fine in Theory

