FH MÜNSTER
University of Applied Sciences

Content-Type: multipart/oracle

Tapping into Format Oracles in Email End-to-End Encryption

FABIAN ISING[1,2], DAMIAN PODDEBNIAK[1], TOBIAS KAPPERT[1],
CHRISTOPH SAATJOHANN[1,2], SEBASTIAN SCHINZEL[1,2]

EMAIL: F.ISING@FH-MUENSTER.DE
MASTODON: @MURGI@INFOSEC.EXCHANGE

[1] *FH Münster University of Applied Sciences*
[2] *National Research Center for Applied Cybersecurity ATHENE*

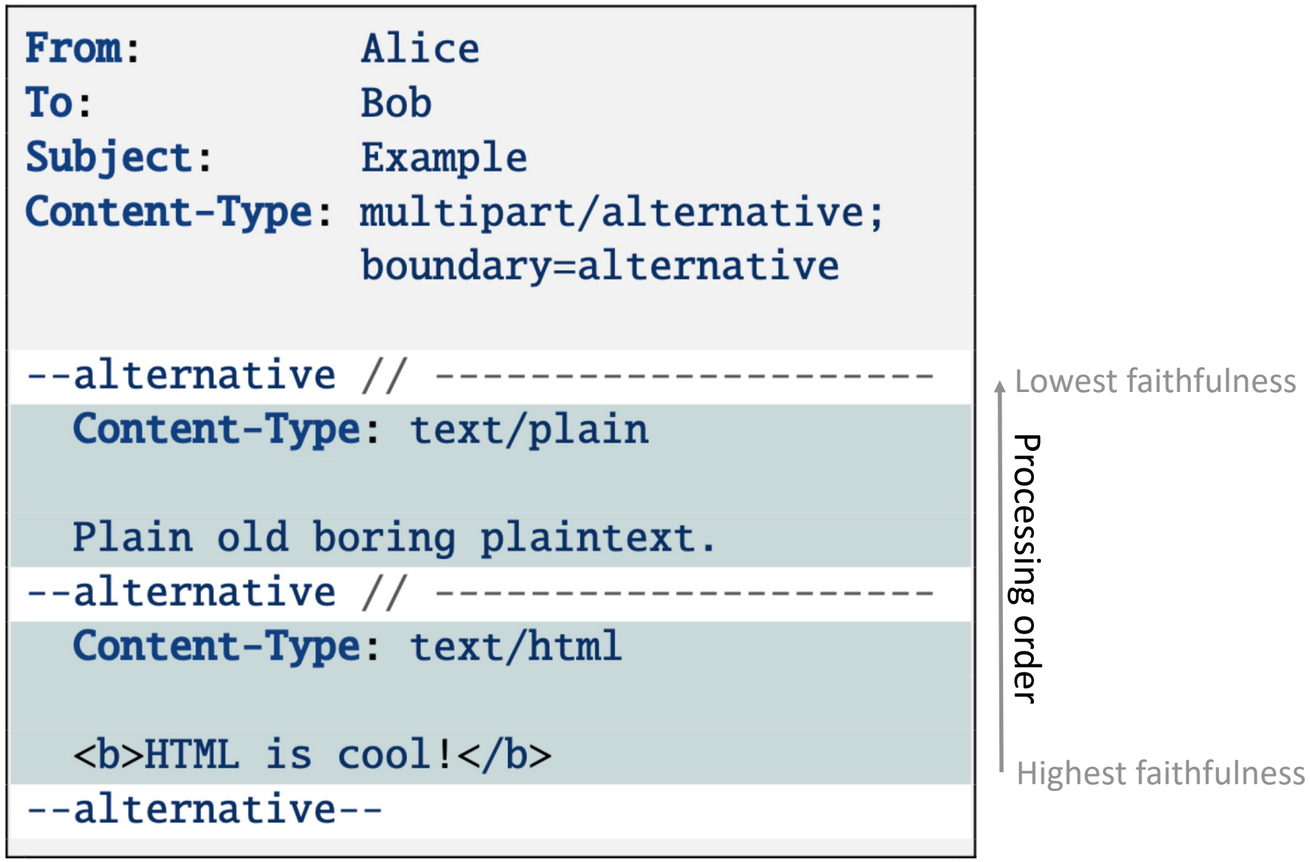
What's an email?

MIME


```
From:          Alice
To:            Bob
Subject:       Example
Content-Type: text/plain

Hello Bob, how was your weekend?
```

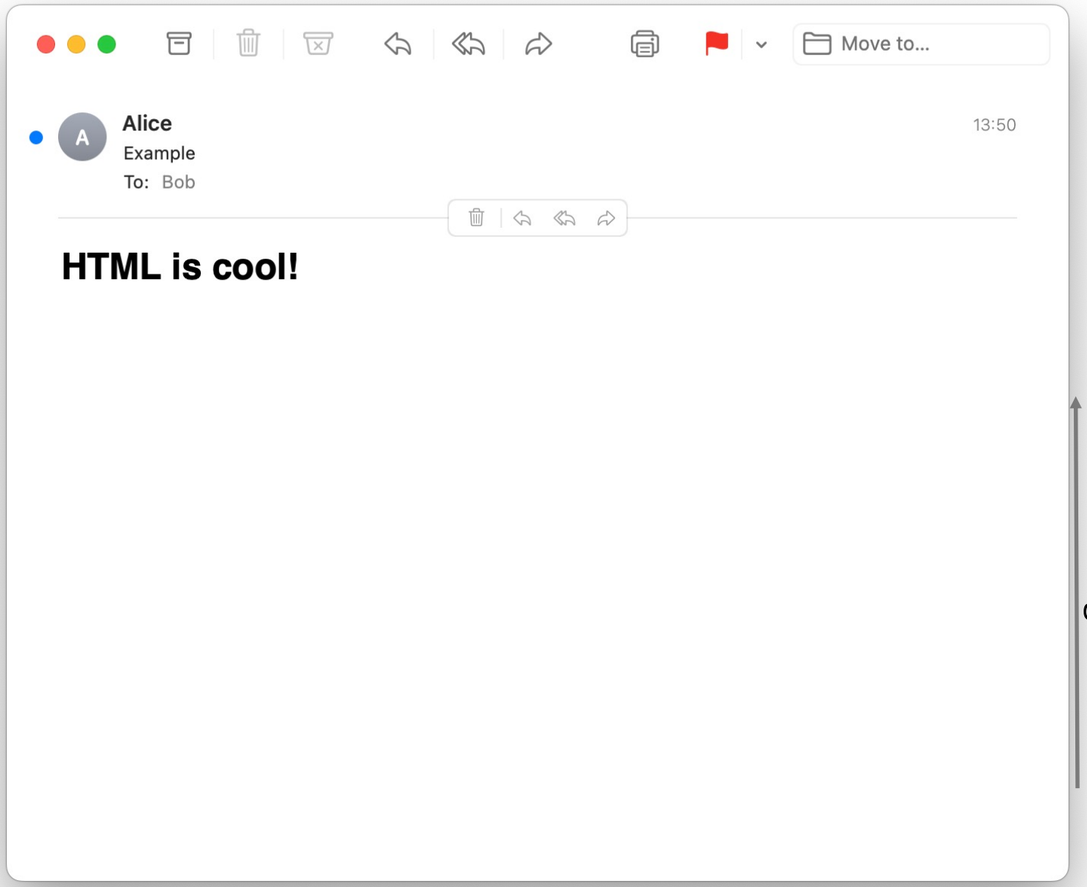
What’s an email?

MIME



What’s an email?

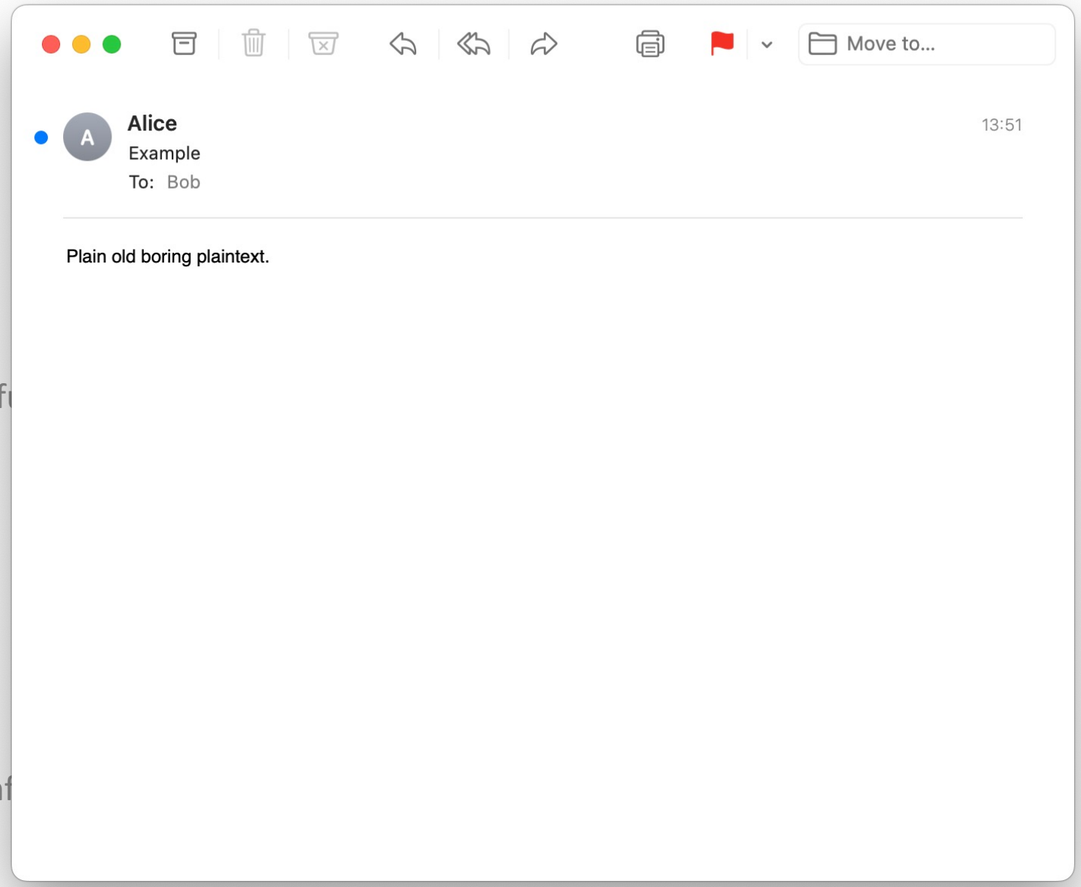
MIME



Lowest faithf

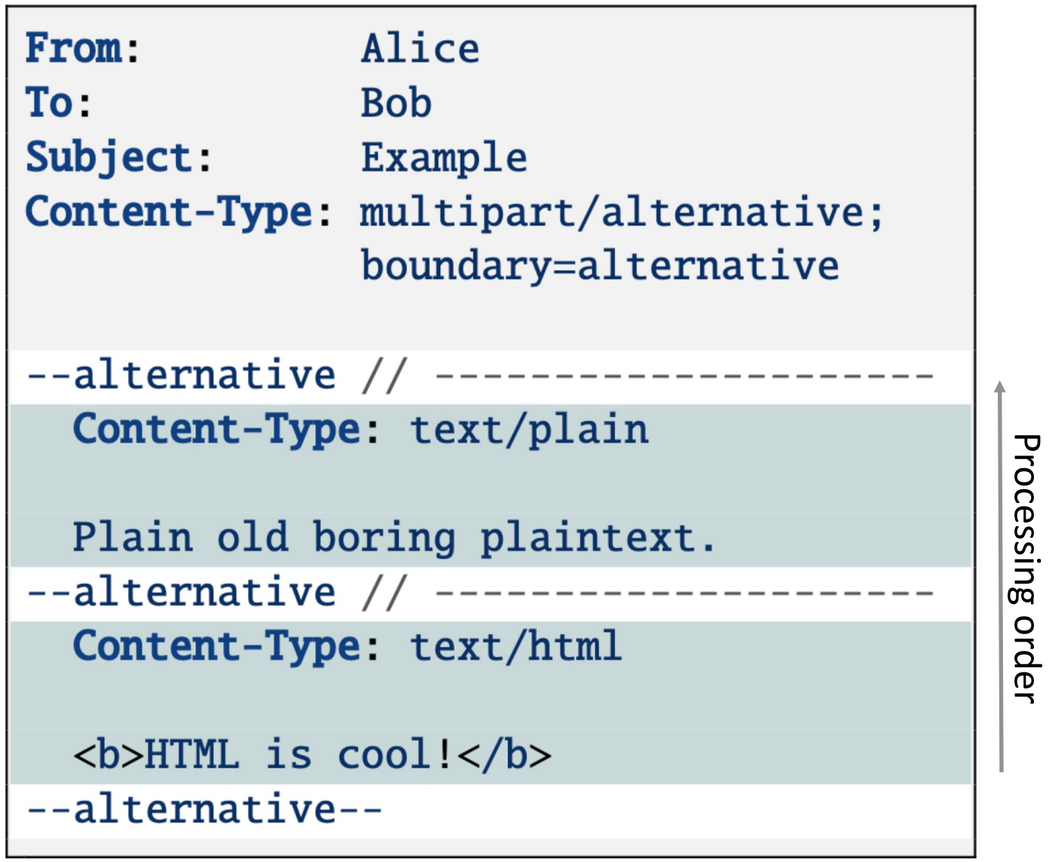
Processing order

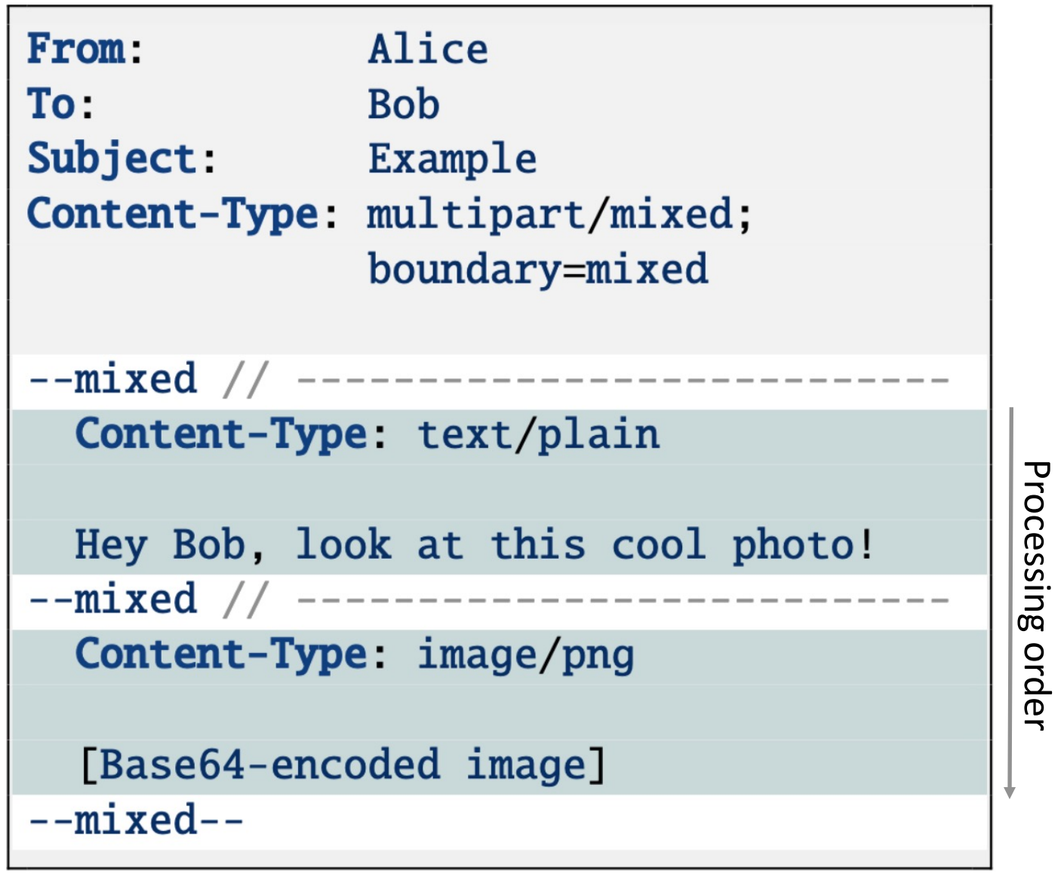
Highest faithf



What's an email?

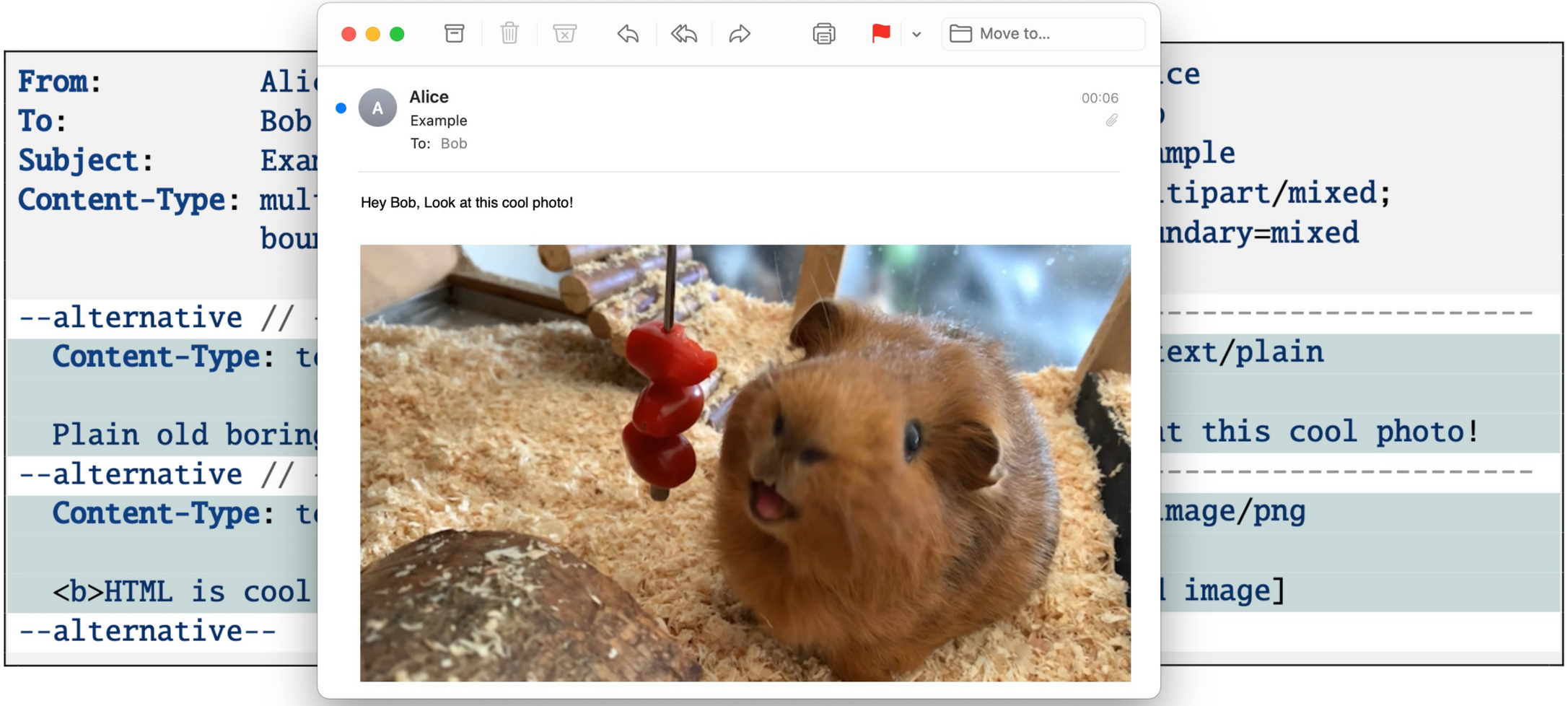
MIME





What’s an email?

MIME



IMAP

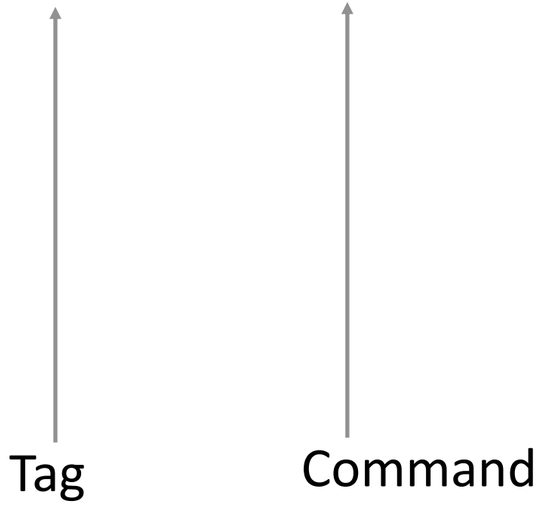
FH MÜNSTER
University of Applied Sciences

```
S: * OK [CAPABILITY IMAP4REV1 AUTH=LOGIN]
```
← Greeting

© Fabian Ising

IMAP


```
S: * OK [CAPABILITY IMAP4REV1 AUTH=LOGIN]
C: A CAPABILITY
```



© Fabian Ising

IMAP

FH MÜNSTER
University of Applied Sciences

```
S: * OK [CAPABILITY IMAP4REV1 AUTH=LOGIN]
C: A CAPABILITY
S: * CAPABILITY IMAP4REV1 AUTH=LOGIN
```

Untagged Response

© Fabian Ising

IMAP

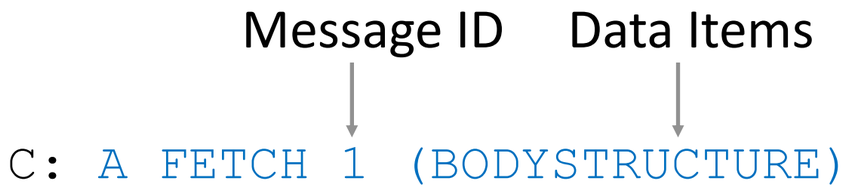
FH MÜNSTER
University of Applied Sciences

```
S: * OK [CAPABILITY IMAP4REV1 AUTH=LOGIN]
C: A CAPABILITY
S: * CAPABILITY IMAP4REV1 AUTH=LOGIN
S: A OK
```

Tagged Response

© Fabian Ising

IMAP




```
From:         Alice
To:           Bob
Subject:      Example
Content-Type: multipart/alternative;
              boundary=alternative

--alternative // ---------------------
  Content-Type: text/plain

  Plain old boring plaintext.
--alternative // ---------------------
  Content-Type: text/html

  <b>HTML is cool!</b>
--alternative--
```

© Fabian Ising

IMAP


```
C: A FETCH 1 (BODYSTRUCTURE)
S: * 1 FETCH (BODYSTRUCTURE (
        ("text" "plain" NIL NIL NIL "7BIT" 29 NIL)
        ("text" "html" NIL NIL NIL "7BIT" 22 NIL)
        "alternative"  ("boundary" "alternative") NIL NIL NIL))
S: A OK fetch done.
C: B FETCH 1 (BODY[1])
S: * 1 FETCH (BODY[1] {29}
        Plain old boring plaintext.
    )
S: B OK fetch done.
```

```
From:         Alice
To:           Bob
Subject:      Example
Content-Type: multipart/alternative;
              boundary=alternative

--alternative // --------------------
  Content-Type: text/plain

  Plain old boring plaintext.
--alternative // --------------------
  Content-Type: text/html

  <b>HTML is cool!</b>
--alternative--
```

© Fabian Ising

Email E2E Encryption

TWO COMPETING STANDARDS

FH MÜNSTER
University of Applied Sciences

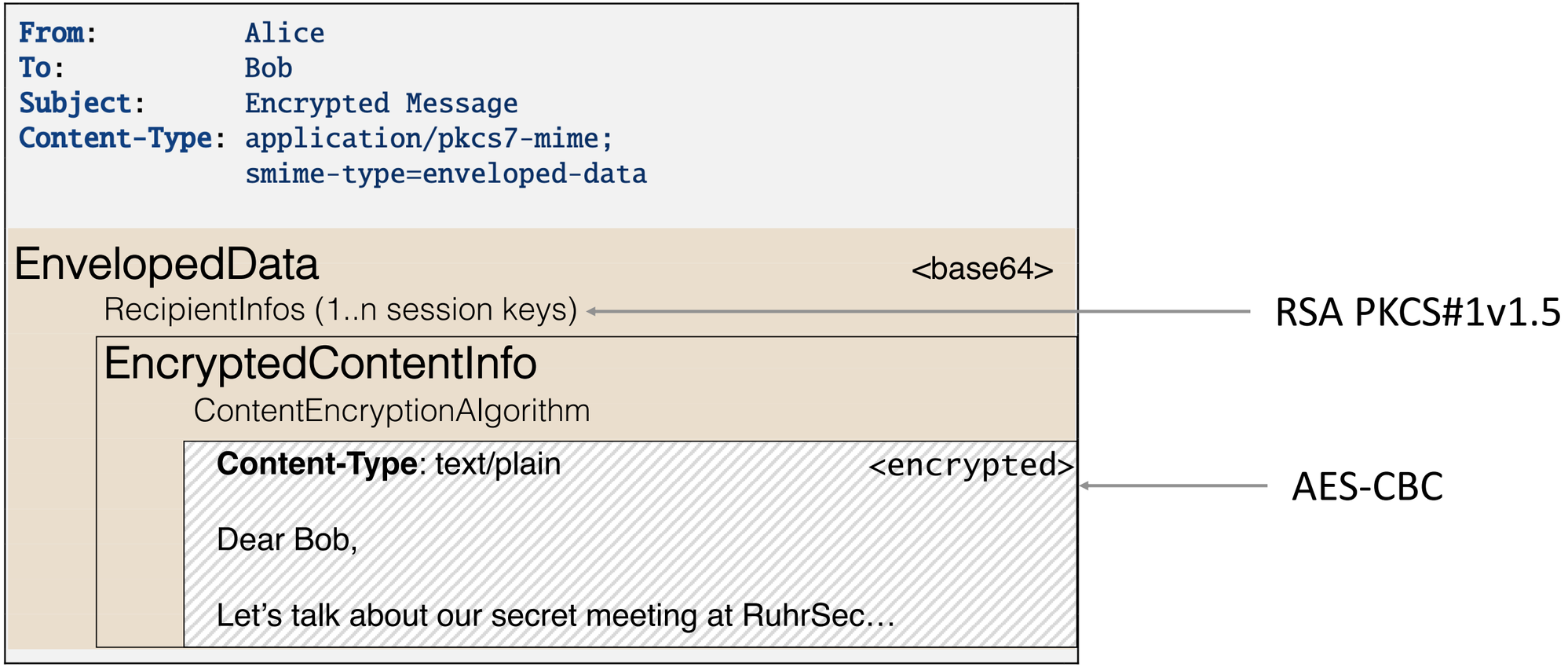
Why?

How?

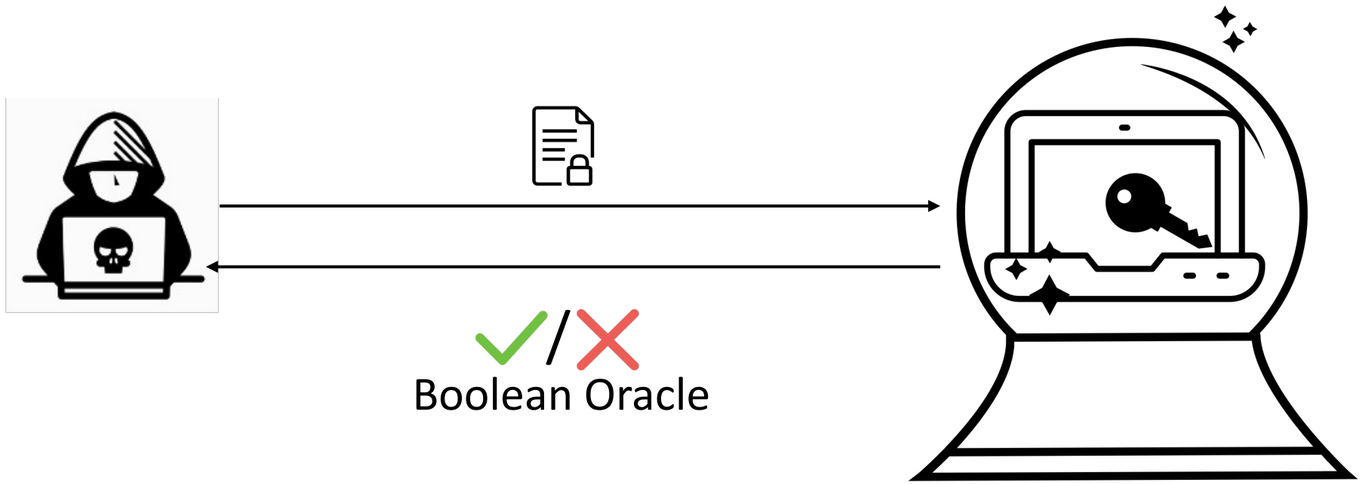
- S/MIME
- OpenPGP

S/MIME

ENCRYPTED MESSAGES

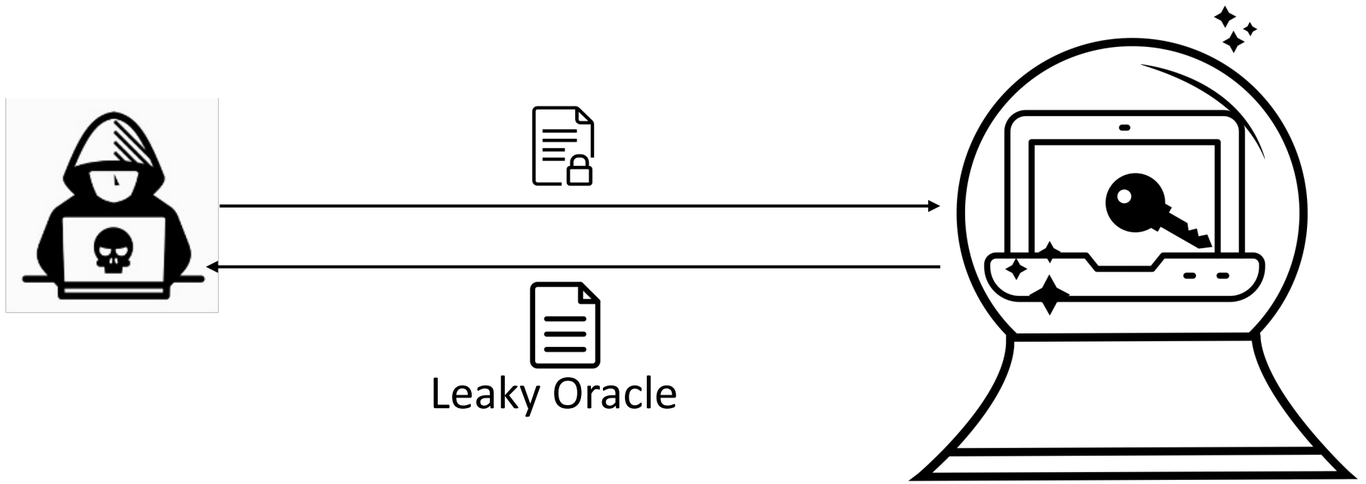


Decryption Oracle Attacks



© Fabian Ising

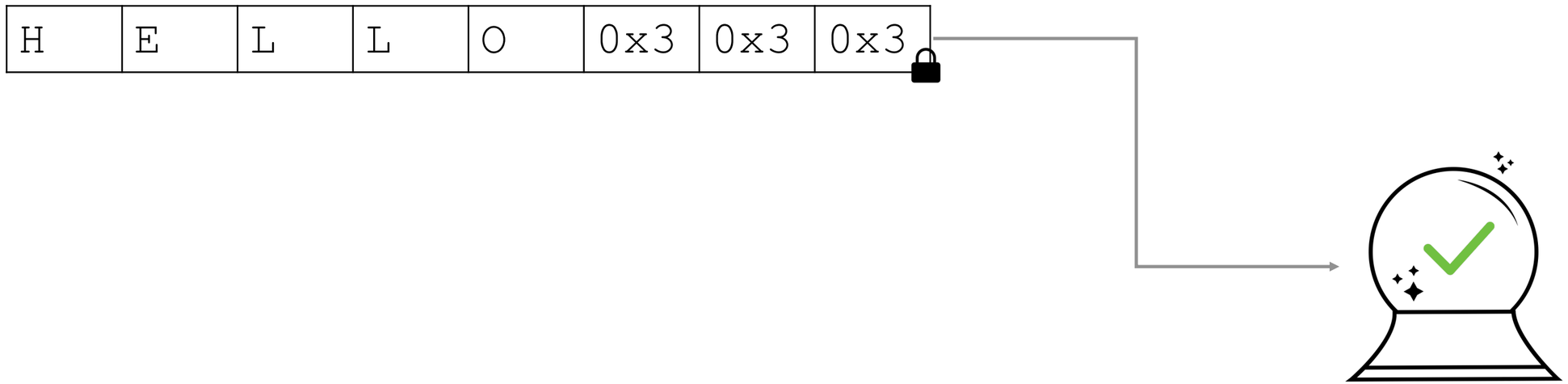
Decryption Oracle Attacks



© Fabian Ising

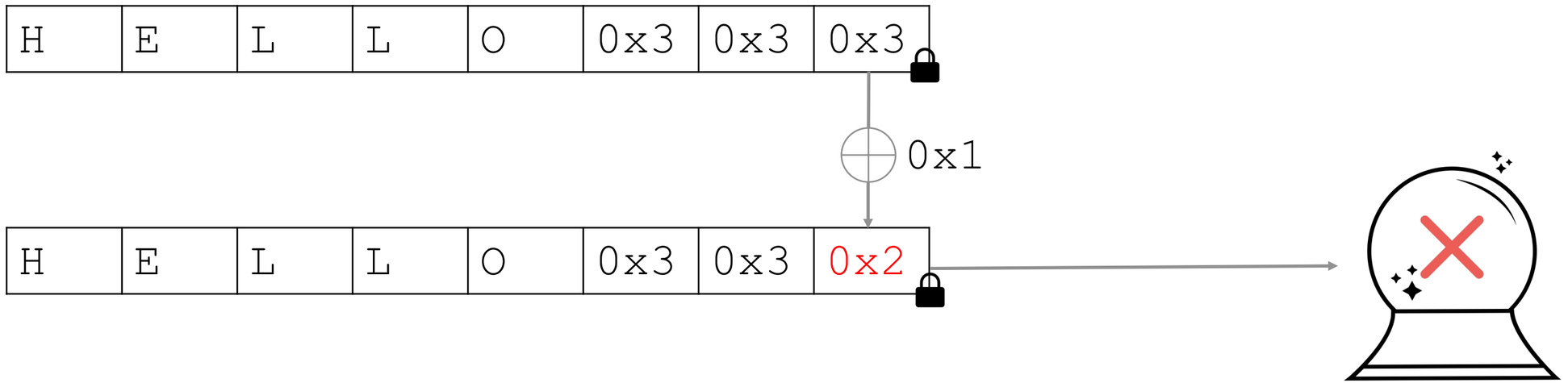
Format Oracles

EXAMPLE: VAUDENAY'S PADDING ORACLE



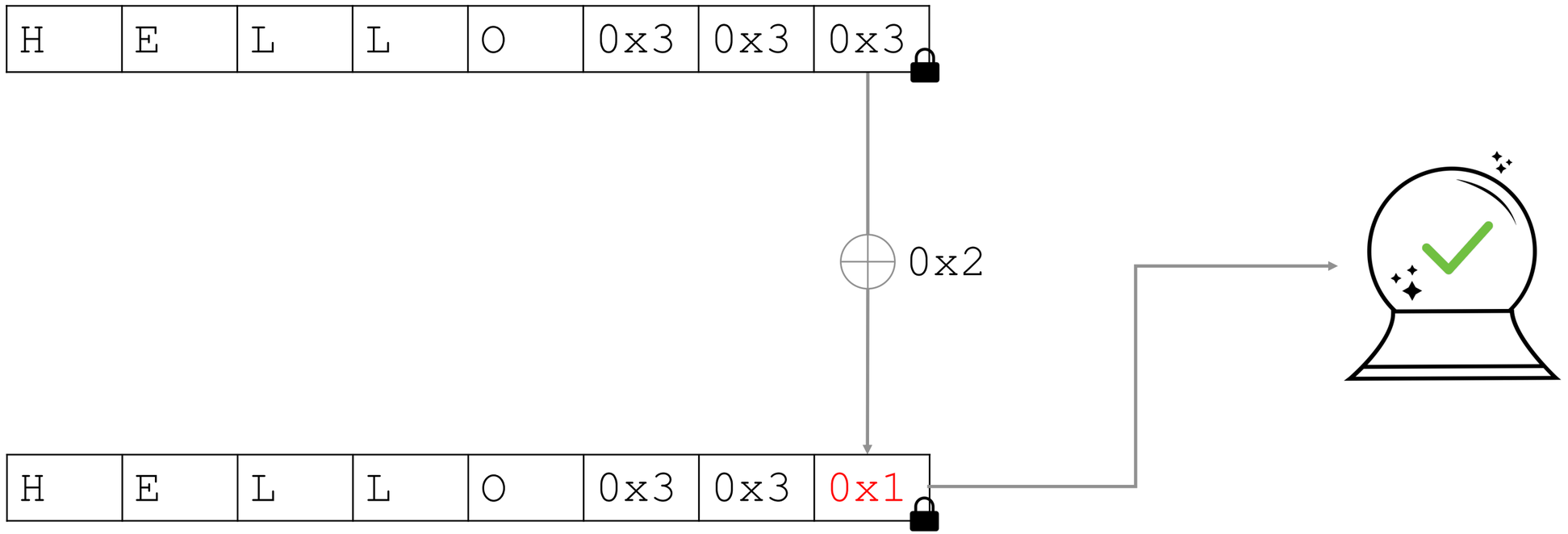
Format Oracles

EXAMPLE: VAUDENAY'S PADDING ORACLE



Format Oracles

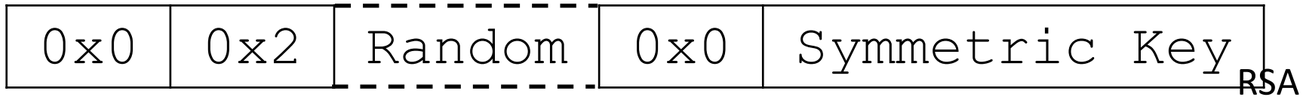
EXAMPLE: VAUDENAY'S PADDING ORACLE

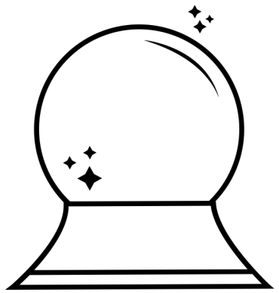


Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

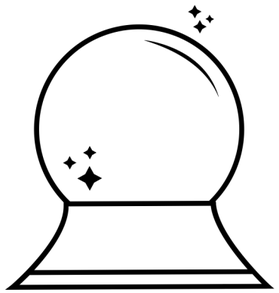
PKCS#1v1.5:





Format Oracles

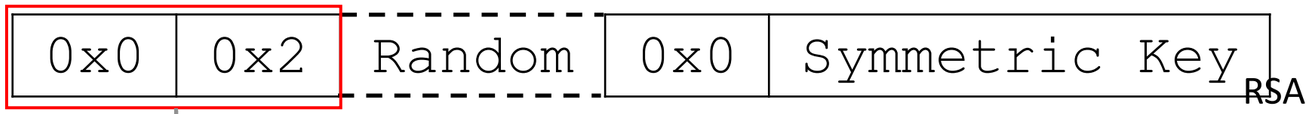
BLEICHENBACHER'S MILLION MESSAGE ATTACK

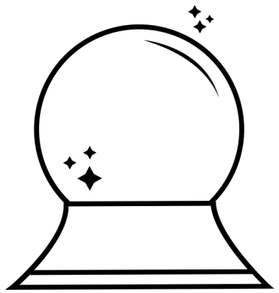


Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

PKCS#1v1.5:





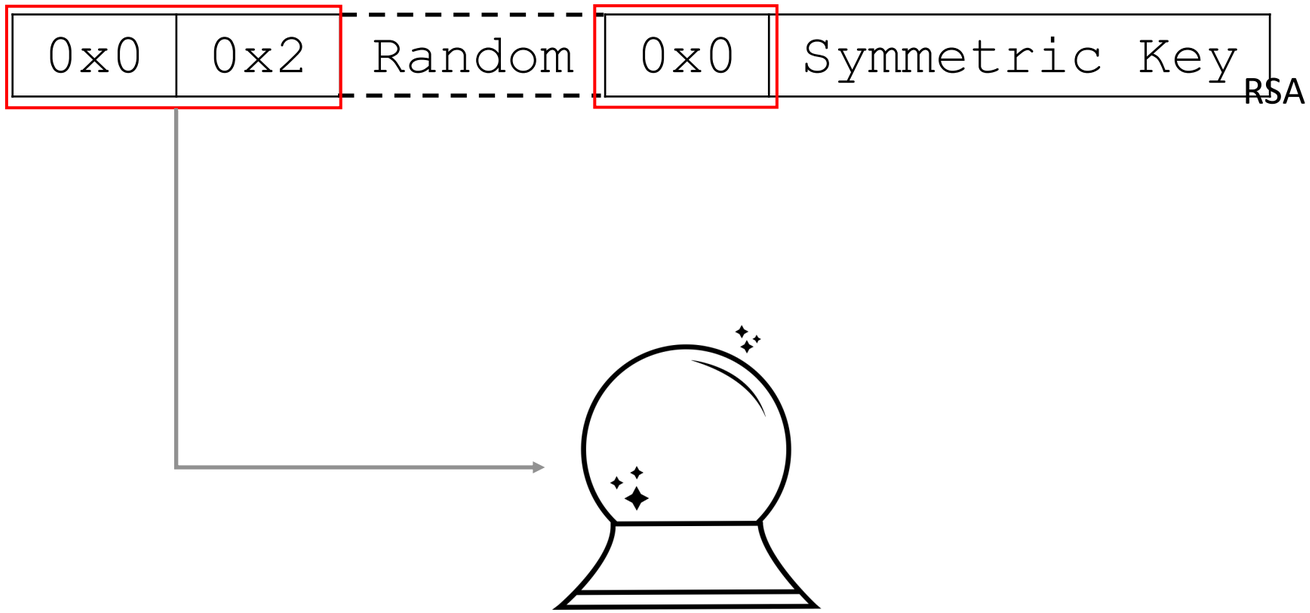
TTT Oracle
~ 10.000 Queries [1]

[1] Mean Query Counts from Bardou et al. – Efficient Padding Oracle Attacks

Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

PKCS#1v1.5:

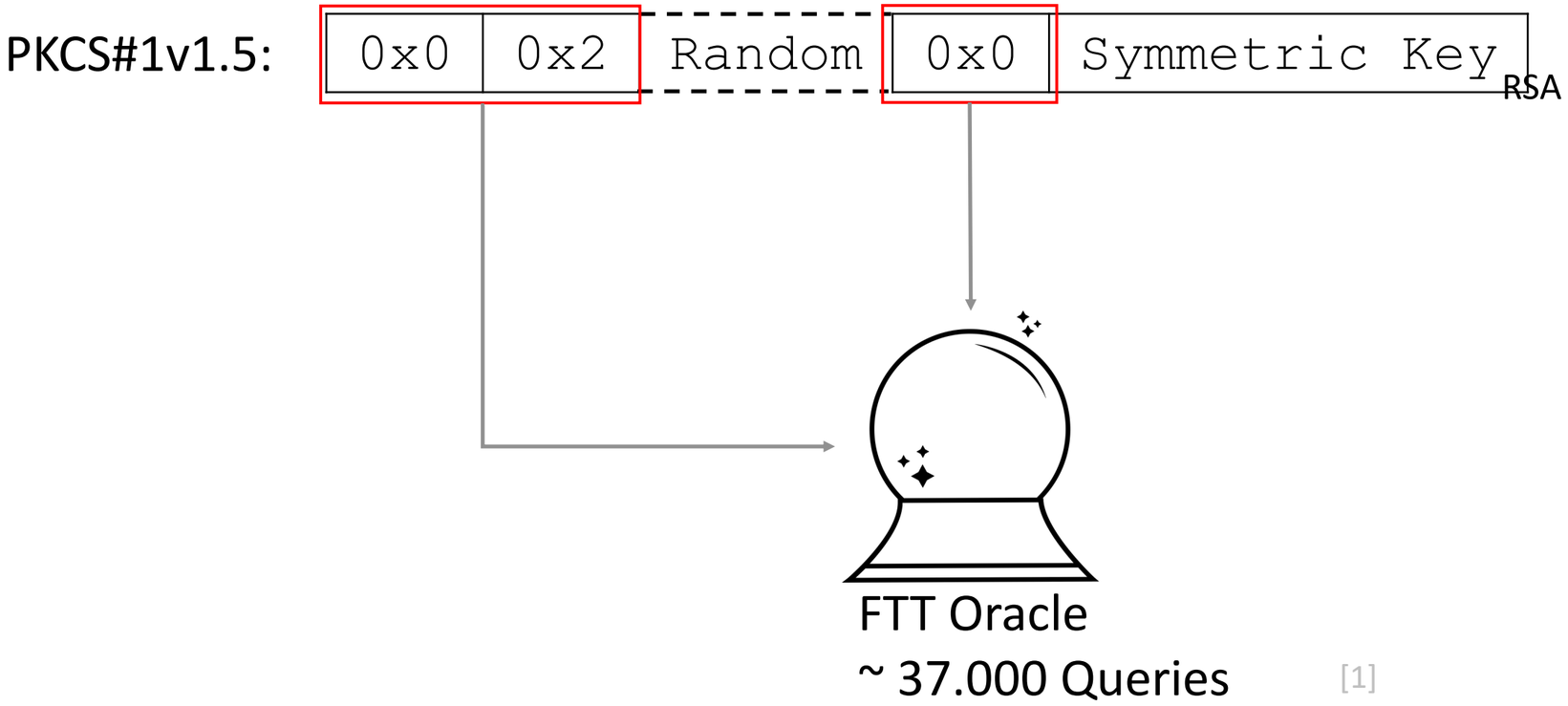


[1]

[1] Mean Query Counts from Bardou et al. – Efficient Padding Oracle Attacks

Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

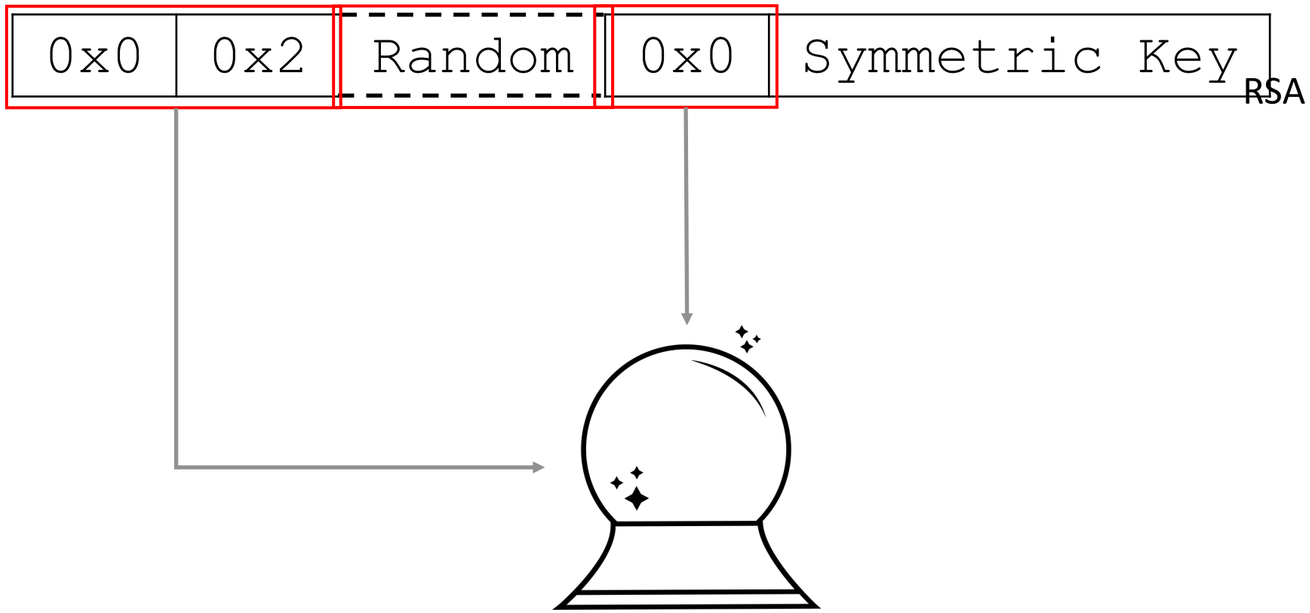


[1] Mean Query Counts from Bardou et al. – Efficient Padding Oracle Attacks

Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

PKCS#1v1.5:



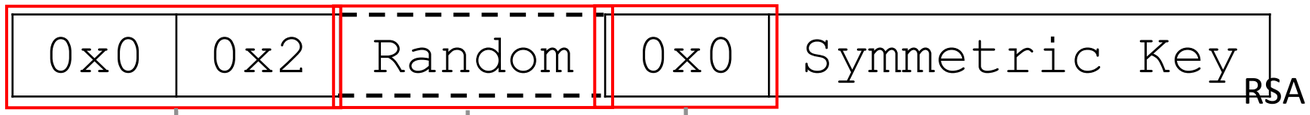
[1]

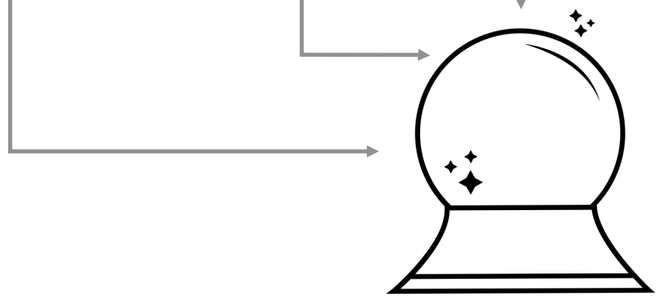
[1] Mean Query Counts from Bardou et al. – Efficient Padding Oracle Attacks

Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

PKCS#1v1.5:





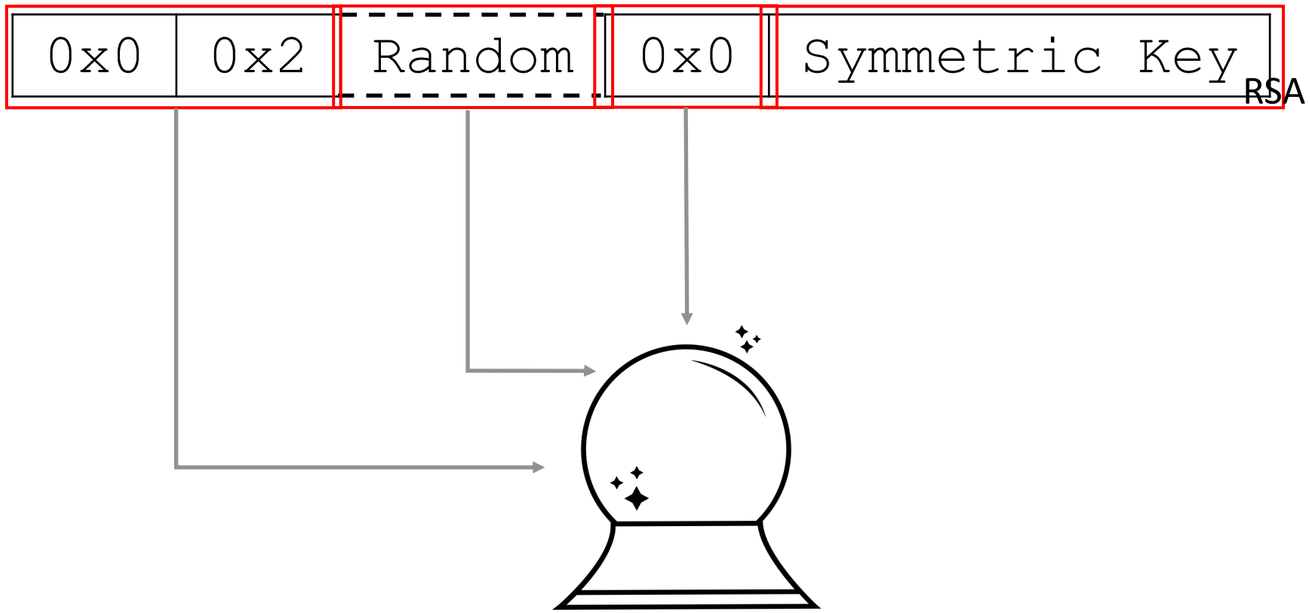
FFT Oracle
~ 50.000 Queries [1]

[1] Mean Query Counts from Bardou et al. – Efficient Padding Oracle Attacks

Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

PKCS#1v1.5:



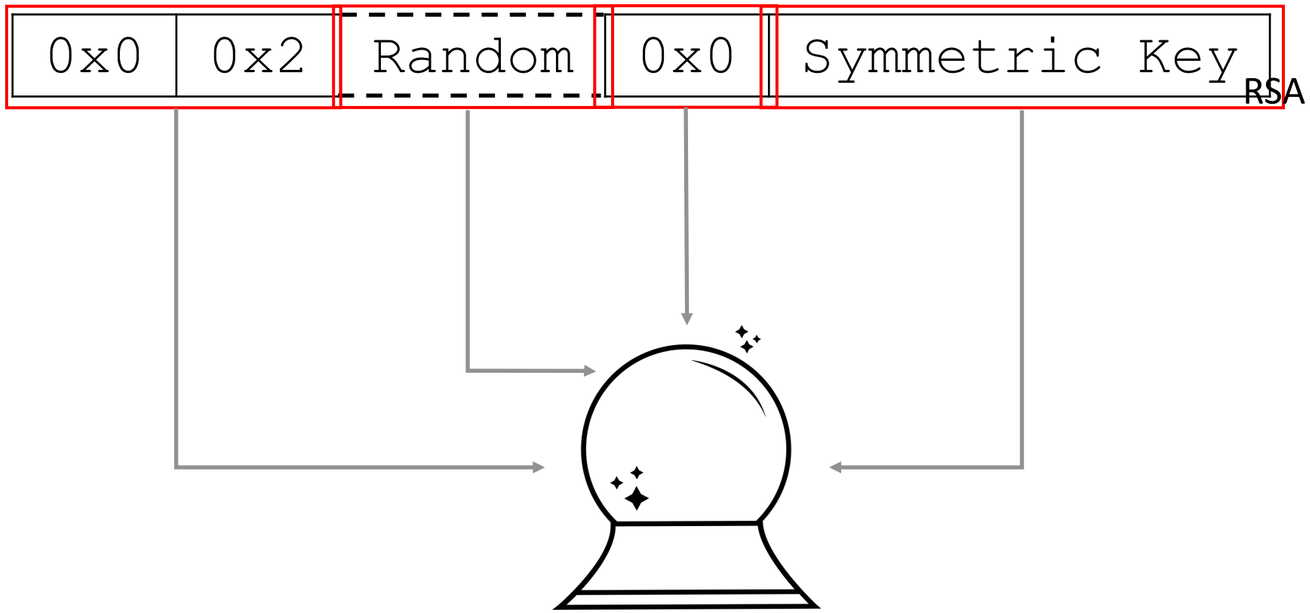
[1]

[1] Mean Query Counts from Bardou et al. – Efficient Padding Oracle Attacks

Format Oracles

BLEICHENBACHER'S MILLION MESSAGE ATTACK

PKCS#1v1.5:



© Fabian Ising

[1] Mean Query Counts from Bardou et al. – Efficient Padding Oracle Attacks

FORMAT ORACLE AGAINST EMAIL E2EE

© Fabian Ising

Previous Work and Feasibility

FH MÜNSTER
University of Applied Sciences

message appears garbled; U therefore replies to A with, for example, "What were you trying to send me?", but also *quotes the "garbled" message* M'. [1]

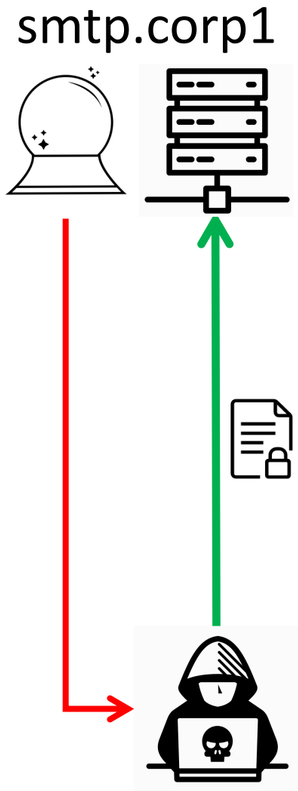
2^{15} oracle queries to determine the first two bytes of any plaintext block, it likely won't effect applications with human end users. Server-based applications would be more vulnerable to attack though. [2]

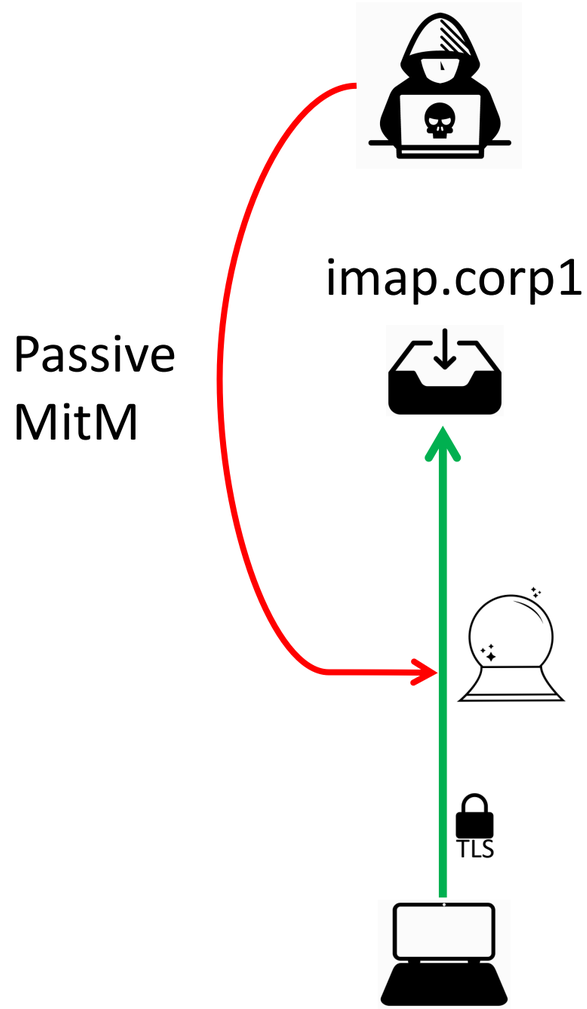
Should the mentioned error messages be made available to an adversary by an inadvertent application, a format oracle would be instantiated. [3]

the HTML rendering engine leaks the decrypted message to the attacker-controlled web server within the URL path of a GET request [4]

[1] Katz, Schneier – A Chosen Ciphertext Attack Against Several E-Mail Encryption Protocols
[2] Mister, Zuccherato – An Attack on CFB Mode Encryption As Used By OpenPGP
[3] Maury et al. – Format Oracles on OpenPGP
[4] Poddebniak et al. – Efail: Breaking S/MIME and OpenPGP Email Encryption using Exfiltration Channels

Attacker Model





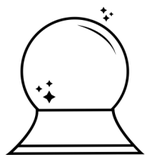
© Fabian Ising

Google Workspaces

THE “EASY” ATTACK

FH MÜNSTER
University of Applied Sciences

```
S: 554 5.7.5 To prevent known S/MIME vulnerabilities,
       Gmail does not accept S/MIME encrypted messages
       without an accompanying valid S/MIME signature.
```



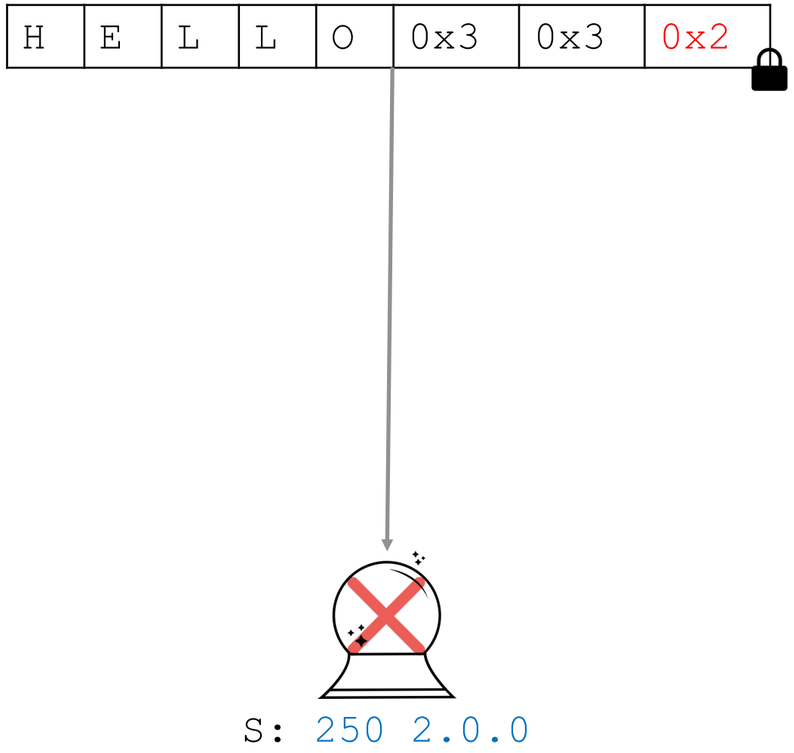
© Fabian Ising

Google Workspaces

THE “EASY” ATTACK

FH MÜNSTER
University of Applied Sciences

```
S: 554 5.7.5 To prevent known S/MIME vulnerabilities,
       Gmail does not accept S/MIME encrypted messages
       without an accompanying valid S/MIME signature.
```



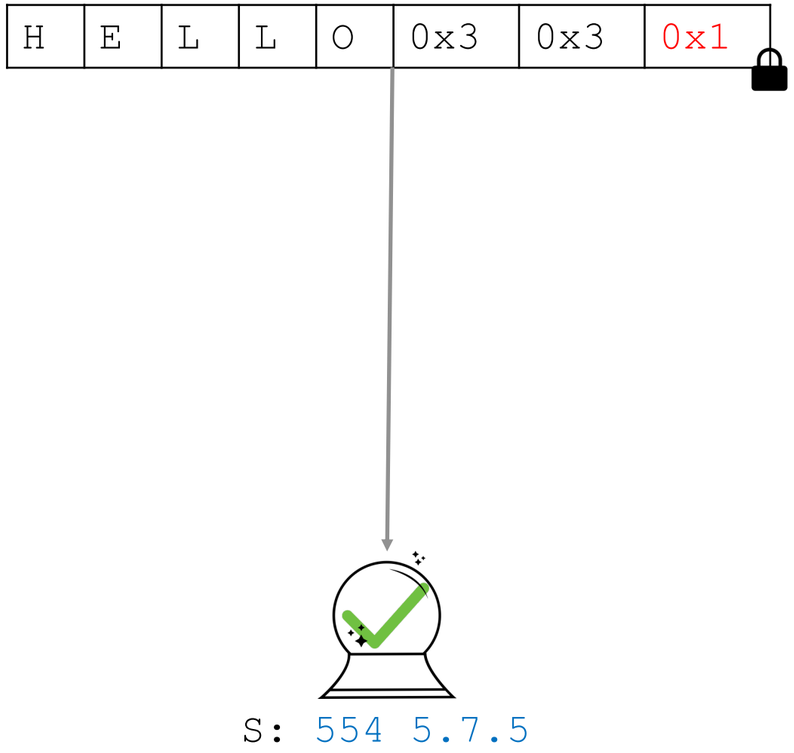
© Fabian Ising

Google Workspaces

THE “EASY” ATTACK

FH MÜNSTER
University of Applied Sciences

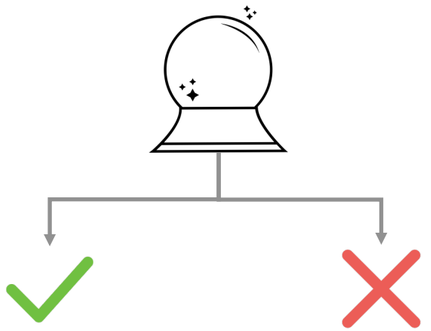
```
S: 554 5.7.5 To prevent known S/MIME vulnerabilities,
       Gmail does not accept S/MIME encrypted messages
       without an accompanying valid S/MIME signature.
```



In Search of Side-Channels


```
From:          Alice
To:            Bob
Subject:       Example
Content-Type:  multipart/alternative;
               boundary=alternative

--alternative // ------------------------------------
  Content-Type: application/encrypted

  [Base64-encoded ciphertext]
--alternative // ------------------------------------
  Content-Type: application/encrypted

  [Base64-encoded ciphertext]
--alternative--
```

```
C: A FETCH 1 (BODYSTRUCTURE)
S: * 1 FETCH (BODYSTRUCTURE (…))
C: B FETCH 1 (BODY[2])
S: * 1 FETCH (BODY[2] {…}
   <Oracle Query>
```



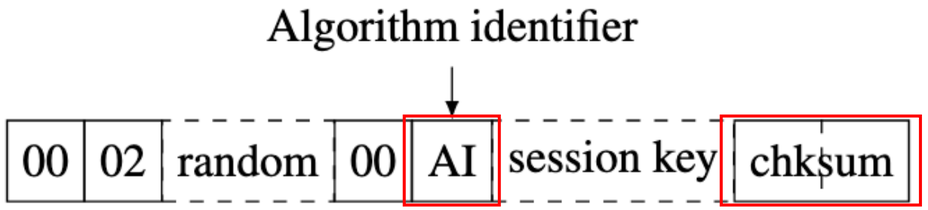
C: C LOGOUT C: C FETCH 1 (BODY[1])

© Fabian Ising

Format Oracles in Email

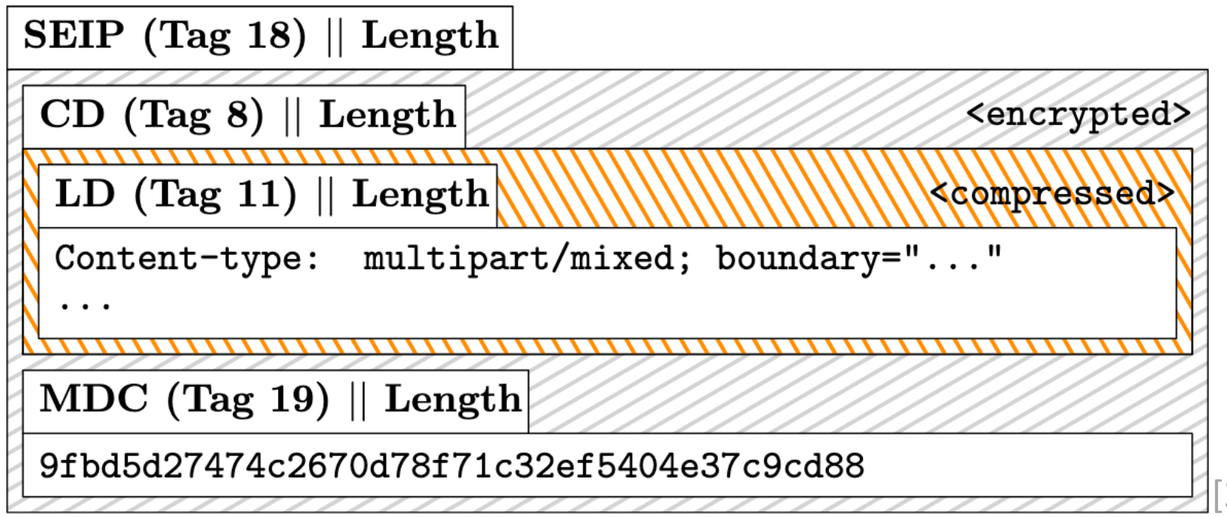
OPENPGP

Million Message Attack:



Attacks on
Symmetric Encryptions:

> – Encrypted data, the output of the selected symmetric-key cipher operating in OpenPGP's variant of Cipher Feedback (CFB) mode. [1]



[2]

[1] RFC 4880
[2] Poddebniak et al. - Efail: Breaking S/MIME and OpenPGP Email Encryption using Exfiltration Channels

Padding Oracles

FH MÜNSTER
University of Applied Sciences

Application / Library	CBC Padding	Million Message Attack	
	Oracle	**Oracle**	**Query Count** [1]
OpenPGP			
GnuPG	–	–	2^{46}
OpenPGP.js	–	–	2^{46}

[1] Worst case query counts based on original algorithm

Padding Oracles

Application / Library	CBC Padding Oracle	Million Message Attack Oracle	Million Message Attack Query Count [1]
OpenPGP			
GnuPG	–	–	2^{46}
OpenPGP.js	–	–	2^{46}
S/MIME			
NSS	N	FFF	2^{26}
GPGME	C	FTF	2^{26}

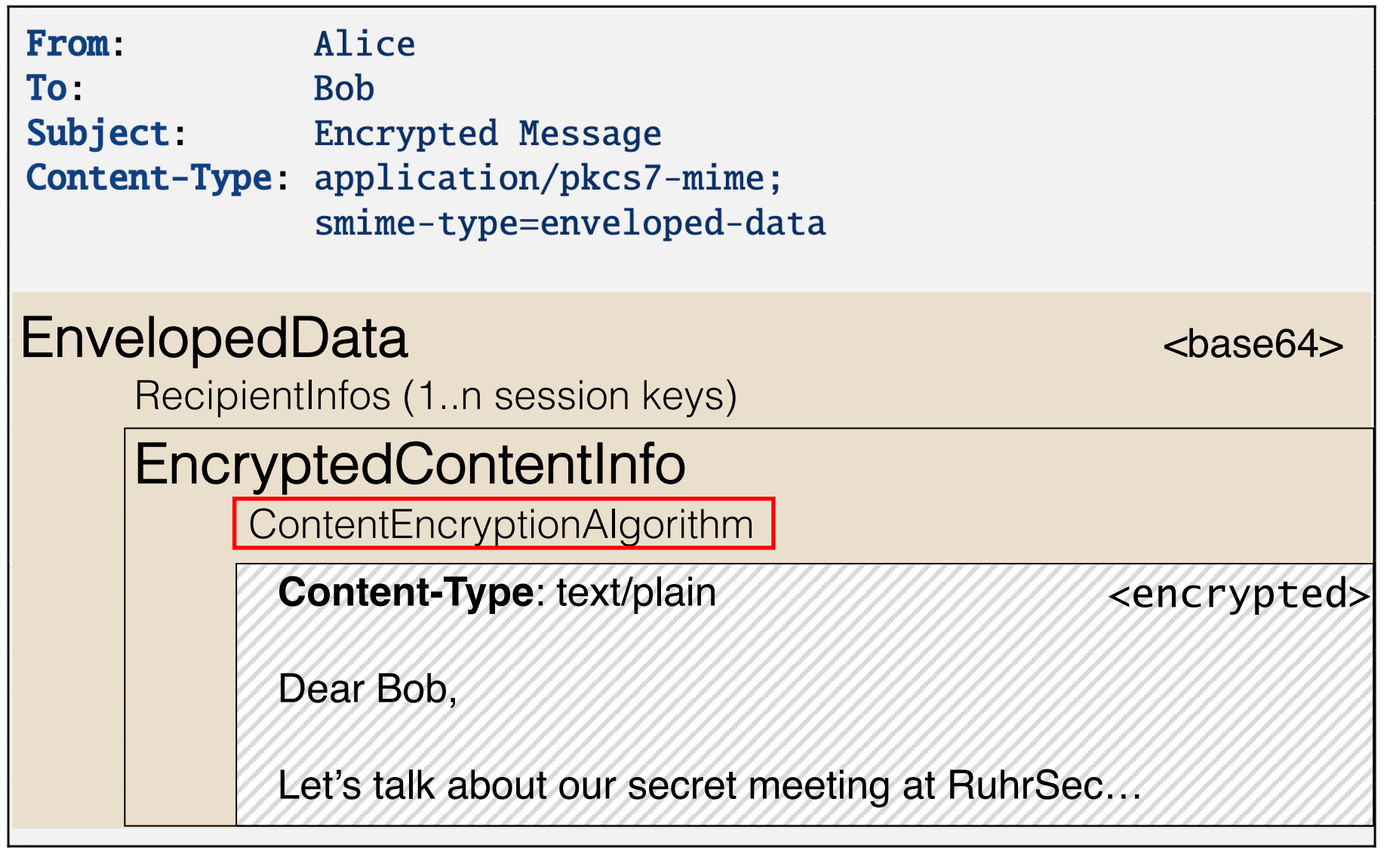
N CBC padding not checked.
C CBC padding checked.
– Not applicable.

© Fabian Ising

[1] Worst case query counts based on original algorithm

S/MIME

ENCRYPTED MESSAGES



© Fabian Ising

Padding Oracles

Application / Library	CBC Padding Oracle	Million Message Attack Oracle	Million Message Attack Query Count [1]
OpenPGP			
GnuPG	–	–	2^{46}
OpenPGP.js	–	–	2^{46}
S/MIME			
NSS	N	FFF	2^{26}
NSS[1]	–	FFT	2^{19}
GPGME	C	FTF	2^{26}
GPGME[1 2]	–	FTT	2^{19}

N CBC padding not checked.
C CBC padding checked.
– Not applicable.

[1] Replacing the algorithm with one with variable key length.
[2] Variable key length algorithms have to be explicitly activated.

© Fabian Ising

[1] Worst case query counts based on original algorithm

	Multiple Encrypted Parts	Fetching Behavior		Decryption	Practical Exploit
		Body (Parts)	Lazy		
Required for practical exploit	✓	●	✓	●	

Legend					
✓	Yes	●	Automatic in background.	■	Found
–	No	○	Needs explicit user interaction.	□	Not found
~	Situation dependent	◑	Upon opening email.		
		?	Not detectable.		

© Fabian Ising

Client	Multiple Encrypted Parts	Fetching Behavior		Decryption	Practical Exploit
		Body (Parts)	Lazy		
Required for practical exploit	✓	●	✓	●	
Clients not supporting multiple encrypted parts					
Airmail	–	●	–	◑	□
eM Client	–	◑	✓	?	□
Mail (macOS)	–	●	~	●	□
MailDroid	–	●	✓	○	□
Nine	–	●	–	◑	□
Outlook 2016	–	●	–	?	□
Outlook 2019	–	●	–	?	□
Postbox	–	●	–	?	□
R2Mail2	–	◑	✓	◑	□
Thunderbird	–	●	–	?	□

© Fabian Ising

Client	Multiple Encrypted Parts	Fetching Behavior		Decryption	Practical Exploit
		Body (Parts)	Lazy		
Required for practical exploit	✓	●	✓	●	
Clients not automatically fetching single body parts					
Claws	✓	◑	–	◑	□
Horde IMP	✓	◑	✓	◑	□
Evolution	✓	◑	–	◑	□
KMail	✓	◑	–	○	□
Mutt	✓	◑	–	○	□
The Bat!	✓	◑	–	○	□
Trojitá	✓	◑	✓	◑	□

© Fabian Ising

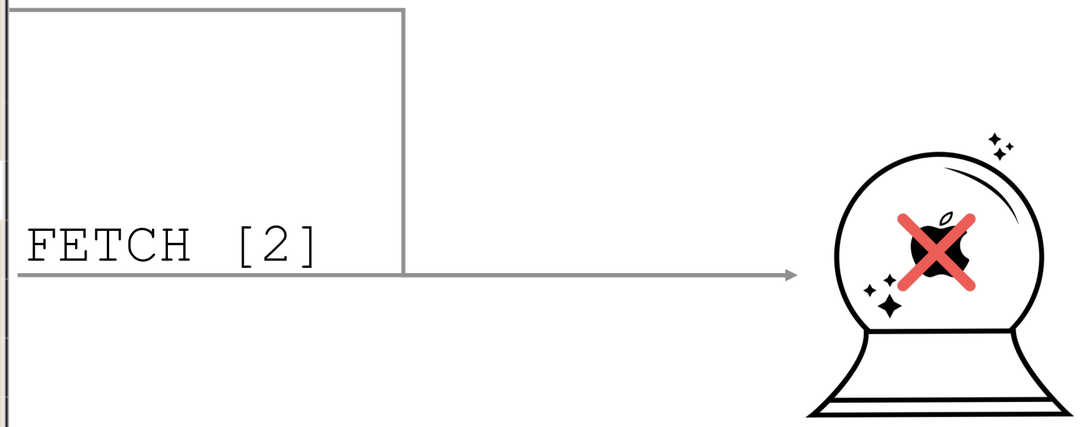
Client	Multiple Encrypted Parts	Fetching Behavior		Decryption	Practical Exploit
		Body (Parts)	Lazy		
Required for practical exploit	✓	●	✓	●	
Clients not using lazy fetching					
MailMate	✓	●	–	◑	□
Clients fulfilling all criteria					
Mail (iOS)	✓	●	✓	●	■

© Fabian Ising

```
Content-Type: multipart/mixed;
              boundary=mixed

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]
// [...]
--mixed--
```

FETCH [1]

```
Content-Type: multipart/mixed;
              boundary=mixed

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]
// [...]
--mixed--
```

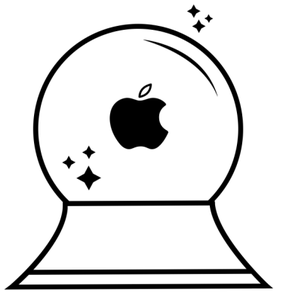


```
Content-Type: multipart/mixed;
              boundary=mixed

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]

--mixed // ---------------------------------
  Content-Type: application/pkcs7-mime;
                smime-type=enveloped-data

  [Base64-encoded S/MIME encrypted message]
// [...]
--mixed--
```

FETCH [1]

FETCH [2]

~~FETCH [3]~~

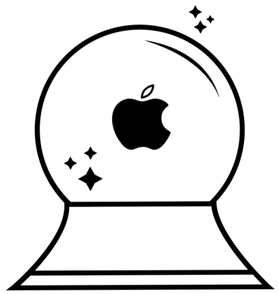


Empty Line Oracle

A NEW FORMAT ORACLE

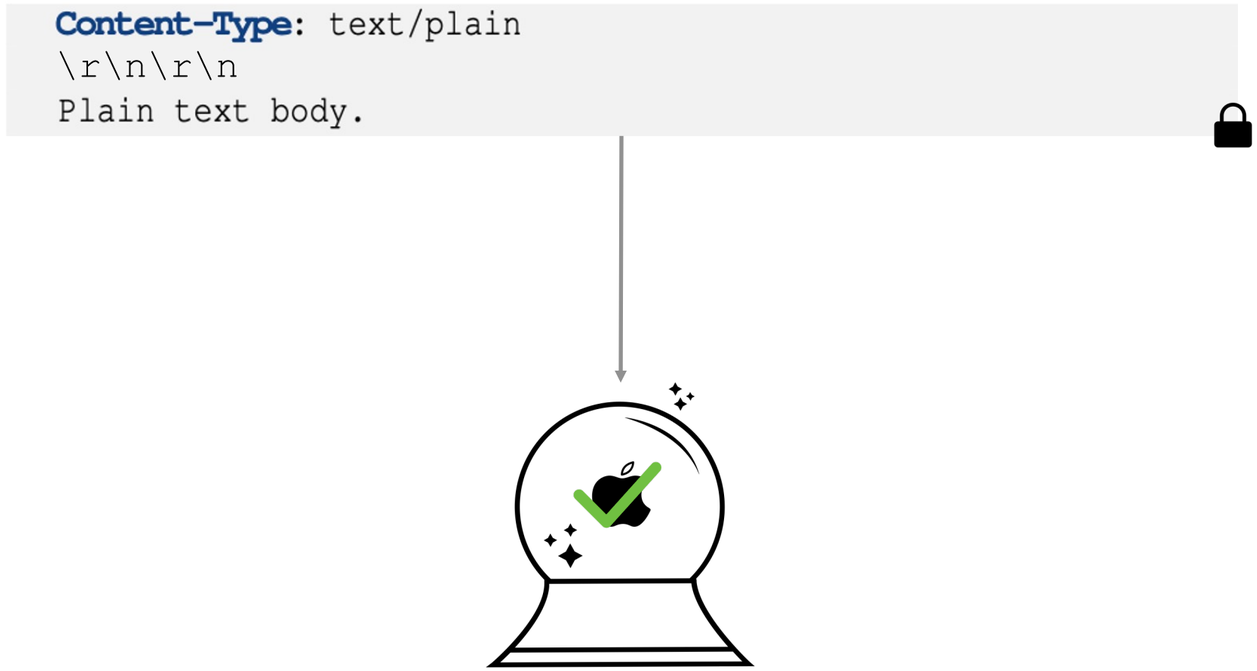
Empty Line Oracle

A NEW FORMAT ORACLE



Empty Line Oracle

A NEW FORMAT ORACLE

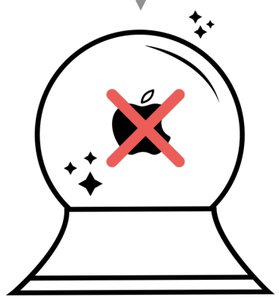


Empty Line Oracle

A NEW FORMAT ORACLE

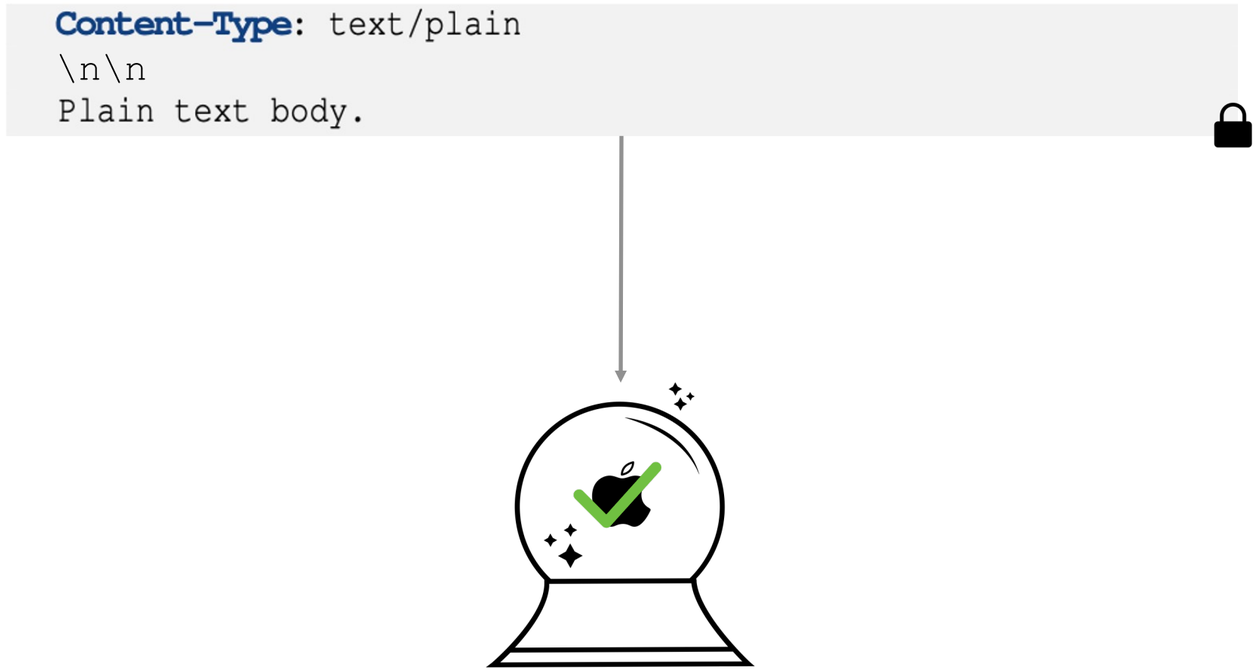
FH MÜNSTER
University of Applied Sciences

```
Content-Type: text/plain
No Empty Line
Plain text body.
```



Empty Line Oracle

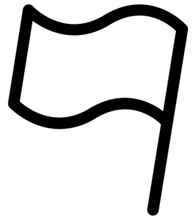
A NEW FORMAT ORACLE



The Empty Line Oracle Exploit

DECRYPTING A BLOCK

FH MÜNSTER
University of Applied Sciences



Setup:

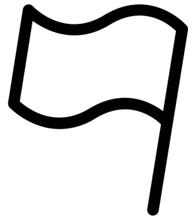
Guess two bytes in a block.

```
\x00    \x00 ✗
\x00    \x01 ✗
  ⋮
\n      \n   ✓
```

The Empty Line Oracle Exploit

DECRYPTING A BLOCK

FH MÜNSTER
University of Applied Sciences



Setup:
Guess two bytes in a block.

Per Byte:
Fixate one known byte to `\n`

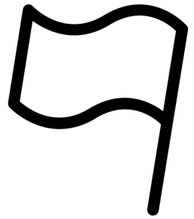
`\n`	`\x00`	✗
`\n`	`\x01`	✗
⋮		
`\n`	`\n`	✓

© Fabian Ising

The Empty Line Oracle Exploit

DECRYPTING A BLOCK

FH MÜNSTER
University of Applied Sciences



Setup:
Guess two bytes in a block.

Per Byte:
Fixate one known byte to `\n`

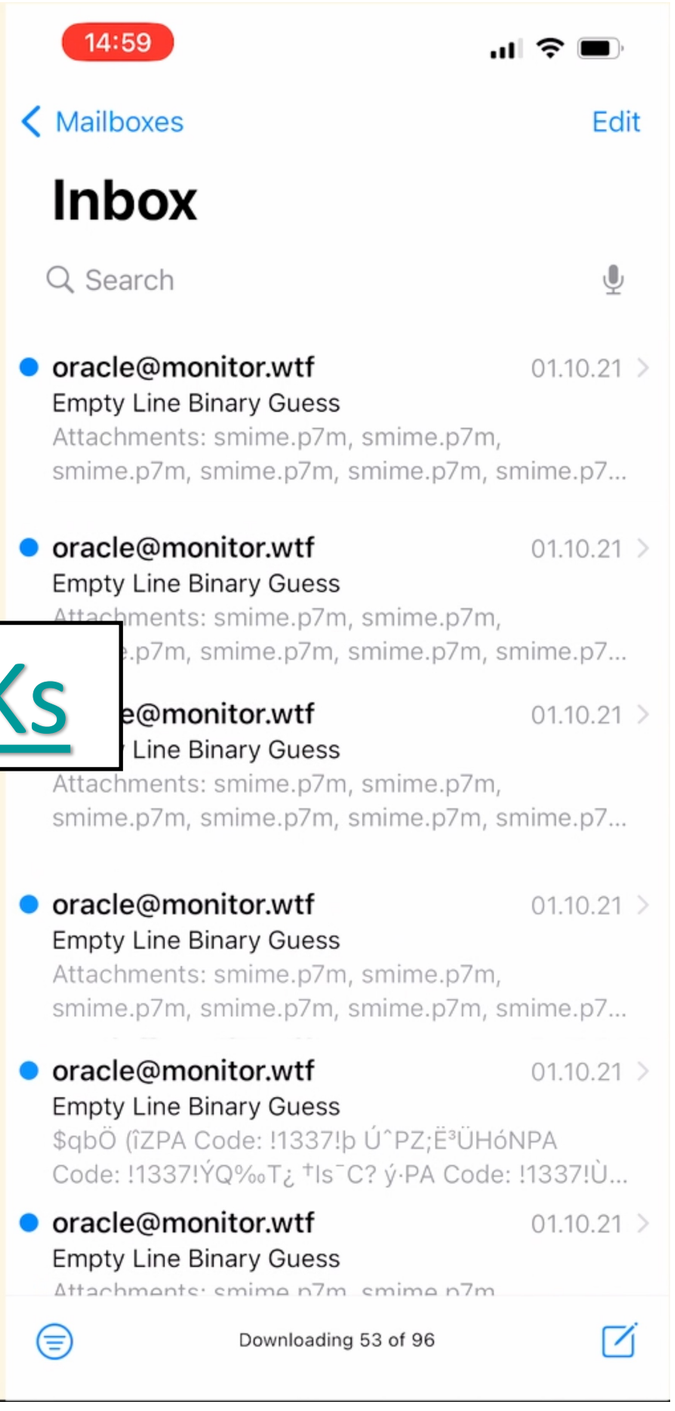
Result:
Decrypted Ciphertext Block

© Fabian Ising

```
2022-12-07 14:59:22,462 - INFO - UID 200104 Using Alphabet b'1\x00'..b'1\x0f'
2022-12-07 14:59:22,463 - INFO - UID 200105 Using Alphabet b'1\x10'..b'1\x1f'
2022-12-07 14:59:22,465 - INFO - UID 200106 Using Alphabet b'1 '..b'1/'
2022-12-07 14:59:22,468 - INFO - UID 200107 Using Alphabet b'10'..b'1?'
2022-12-07 14:59:22,472 - INFO - UID 200108 Using Alphabet b'1@'..b'1O'
2022-12-07 14:59:22,476 - INFO - UID 200109 Using Alphabet b'1P'..b'1_'
2022-12-07 14:59:22,477 - INFO - UID 200110 Using Alphabet b'1`'..b'1o'
2022-12-07 14:59:22,479 - INFO - UID 200111 Using Alphabet b'1p'..b'1\x7f'
2022-12-07 14:59:22,484 - INFO - UID 200112 Using Alphabet b'2\x00'..b'2\x0f'
2022-12-07 14:59:22,488 - INFO - UID 200113 Using Alphabet b'2\x10'..b'2\x1f'
2022-12-07 14:59:22,493 - INFO - UID 200114 Using Alphabet b'2 '..b'2/'
2022-12-07 14:59:22,497 - INFO - UID 200115 Using Alphabet b'20'..b'2?'
2022-12-07 14:59:22,499 - INFO - UID 200116 Using Alphabet b'2@'..b'2O'
2022-12-07 14:59:22,501 - INFO - UID 200117 Using Alphabet b'2P'..b'2_'
2022-12-07 14:59:22,503 - INFO - UID 200118 Using Alphabet b'2`'..b'2o'
2022-12-07 14:59:22,506 - INFO - UID 200119 Using Alphabet b'2p'..b'2\x7f'
2022-12-07 14:59:22,509 - INFO - UID 200120 Using Alphabet b'3\x00'..b'3\x0f'
2022-12-07 14:59:22,512 - INFO - UID 200121 Using Alphabet b'3\x10'..b'3\x1f'
2022-12-07 14:59:22,513 - INFO - UID 200122 Using Alphabet b'3 '..b'3/'
2022-12-07 14:59:22,516 - INFO - UID 200123 Using Alphabet b'30'..b'3?'
2022-12-07 14:59:22,520 - INFO - UID 200124 Using Alphabet b'3@'..b'30'
2022-12-07 14:59:22,523 - INFO - UID 20
2022-12-07 14:59:22,527 - INFO - UID 20
2022-12-07 14:59:22,531 - INFO - UID 20
```

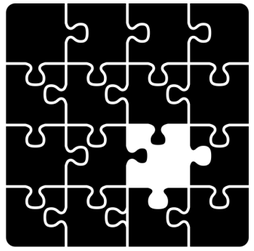
https://youtu.be/1_nvhFfWVKs

```
59D12Jwp | .. Kh/FIKPKLomw6O5+R85x3APmHbzMHHV2xSNNLS3ob+wqH8Ugo8ouibDo7n5HznHcA+YdvMwcdXbF\n
59D12Jwp | .. I0ItLehv7CofxSCjyi6JsOjufkfOcdwD5h28zBx1dsUjQy0t6G/sKh/FIKPKLomw6O5+R85x3APm\n
59D12Jwp | .. HbzMHHV2xSNALS3ob+wqH8Ugo8ouibDo7n5HznHcA+YdvMwcdXbFI0EtLehv7CofxSCjyi6JsOju\n
59D12Jwp | .. fkfOcdwD5h28zBx1dsUjRi0t6G/sKh/FIKPKLomw6O5+R85x3APmHbzMHHV2xSNHLS3ob+wqH8Ug\n
59D12Jwp | .. o8ouibDo7n5HznHcA+YdvMwcdXbFI0QtLehv7CofxSCjyi6JsOjufkfOcdwD5h28zBx1dsUjRS0t\n
59D12Jwp | .. 6G/sKh/FIKPKLomw6O5+R85x3APmHbzMHHV2xSNaLS3ob+wqH8Ugo8ouibDo7n5HznHcA+YdvMwc\n
59D12Jwp | .. dXbFI1stLehv7CofxSCjyi6JsOjufkfOcdwD5h28zBx1dsUjWC0t6G/sKh/FIKPKLomw6O5+R85x\n
59D12Jwp | .. 3APmHbzMHHV2xSNZLS3ob+wqH8Ugo8ouibDo7n5HznHcA+YdvMwcdXbFI14tLehv7CofxSCjyi6J\n
59D12Jwp | .. sOjufkfOcdwD5h28zBx1dsUjXy0t6G/sKh/FIKPKLomw6O5+R85x3APmHbzMHHV2xSNcLS3ob+wq\n
59D12Jwp | .. H8Ugo8ouibDo7n5HznHcA+YdvMwcdXbFI10tLehv7CofxSCjyi6JsOjufkfOcdwD5h28zBx1dsUj\n
59D12Jwp | .. Ui0t6G/sKh/FIKPKLomw6O5+R85x3APmHbzMHHV2xSNTLS3ob+wqH8Ugo8ouibDo7n5HznHcA+Yd\n
59D12Jwp | .. vMwcdXbFI1AtLehv7CofxSCjyi6JsOjufkfOcdwD5h28zBx1dsUjUS0t6G/sKh/FIKPKLomw6O5+\n
59D12Jwp | .. R85x3APmHbzMHHV2xSNWLS3ob+wqH8Ugo8ouibDo7n5HznHcA+YdvMwcdXbFI1ctLehv7CofxSCj\n
59D12Jwp | .. yi6JsOjufkfOcdwD5h28zBx1dsUjVC0t6G/sKh/FIKPKLomw6O5+R85x3APmHbzMHHV2xSNVLS3o\n
59D12Jwp | .. b+wqH8Ugo8ouibDo7n5HznHcA+YdvMwcdXbF\n
59D12Jwp | .. )\r\n
59D12Jwp | S: 4 OK fetch done.\r\n
59D12Jwp | C: 5 LOGOUT\r\n
59D12Jwp | S: * BYE bye done.\r\n
59D12Jwp | S: 5 OK logout done.\r\n
59D12Jwp | {"message": "Connection was closed."}
```

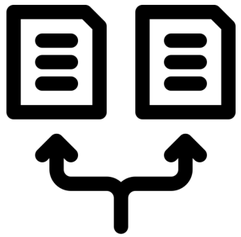


Resistance

WHY ARE THE CLIENTS NOT VULNERABLE?



Incomplete Implementations



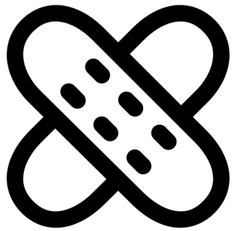
Selective Fetching

Parallel Decryption & Fetching

Implementation Quirks

Countermeasures

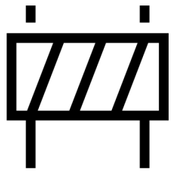
SHORT- AND MID-TERM



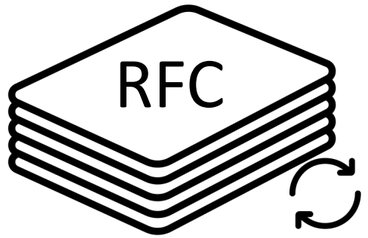
Stopgap

Prevent Oracles

Constant Time Operations



Restrict Features



Update Standards

AEAD

[1]

© Fabian Ising

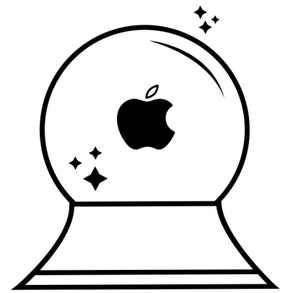
[1] Schwenk et al. - Mitigation of Attacks on Email End-to-End Encryption

Conclusion

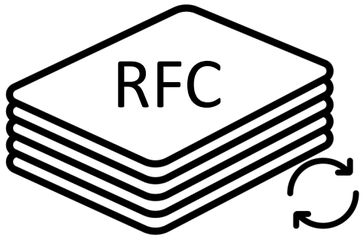
ORACLE ATTACK ON EMAIL E2EE

Many Clients
are safe

'Defenses' are
involuntary



Some Attacks
still possible



Explicit Countermeasures
are necessary

© Fabian Ising

FH MÜNSTER
University of Applied Sciences

Thanks for Listening!

FABIAN ISING

EMAIL: F.ISING@FH-MUENSTER.DE
MASTODON: @MURGI@INFOSEC.EXCHANGE

© Fabian Ising