



声 明

- 本PPT是电子工业出版社出版的教材《计算机网络安全原理》配套教学PPT（部分内容的深度和广度在教材的基础上有所扩展），作者：吴礼发
 - 本PPT可能直接或间接采用了网上资源、公开学术报告中的部分PPT页面、图片、文字，引用时我们力求在该PPT的备注栏或标题栏中注明出处，如果有疏漏之处，敬请谅解。同时对被引用资源或报告的作者表示诚挚的谢意！
 - 本PPT可免费使用、修改，使用时请保留此页。
-

第十章 电子邮件安全





内容提纲

1

电子邮件安全问题

2

安全电子邮件标准**PGP**

3

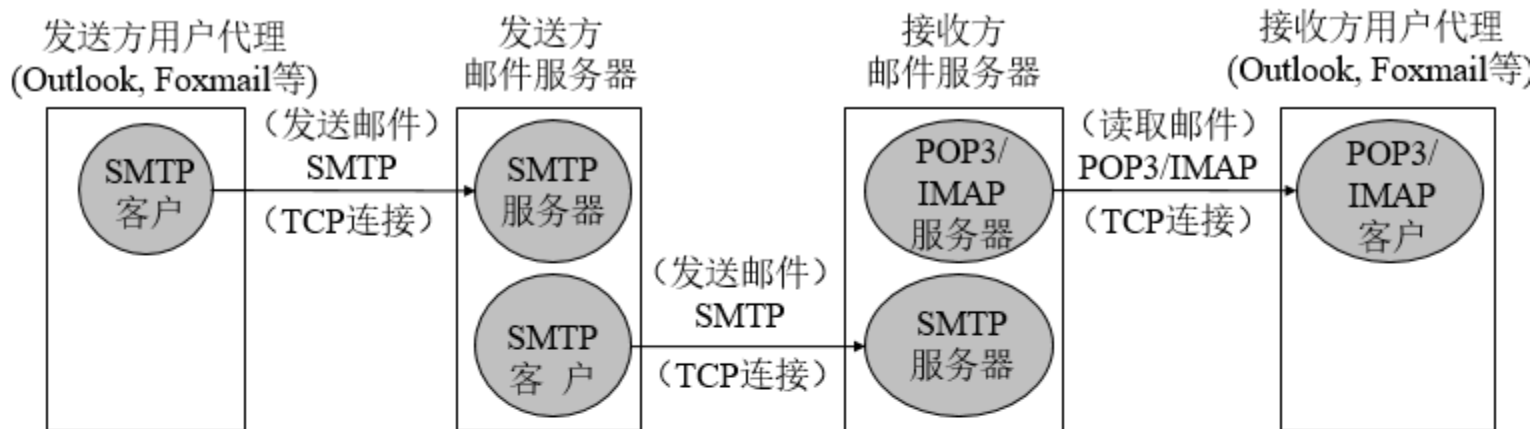
WebMail安全威胁及防范

4

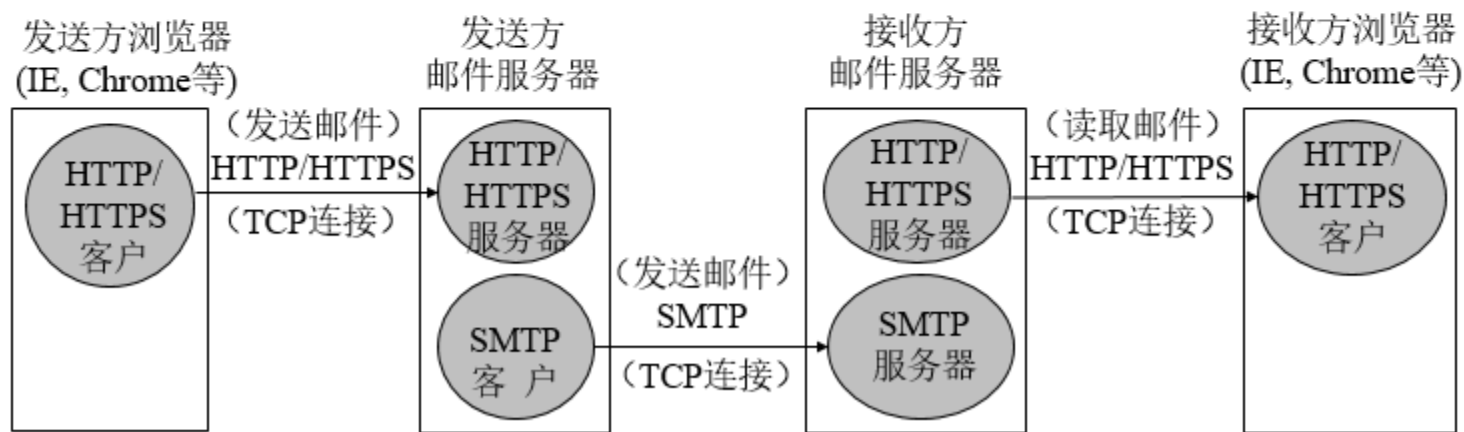
垃圾邮件防范



电子邮件



(a) 用户代理收发邮件



(b) WebMail 收发邮件



电子邮件安全

■ 安全需求

安全需求	说 明
机密性	保证邮件在传输过程中不会被第三方窃取，只有邮件的真正接收方才能够阅读邮件的内容，即使发错邮件，接收方也无法看到邮件内容。
完整性	保证邮件在传输过程中不会被修改
不可否认性	保证邮件发送人不能否认其发过的邮件
真实性	保证邮件的发送人不是冒名顶替的，它同邮件完整性一起可防止攻击者伪造邮件





电子邮件安全

- 基于SMTP、POP3/IMAP等协议的电子邮件系统没有采取必要的安全防护措施，导致：
 - 邮件内容被窃听
 - 垃圾邮件（Spam）
 - 邮件炸弹
 - 传播恶意代码（钓鱼邮件）
 - 电子邮件欺骗





电子邮件安全

- 安全措施：

- **端到端的安全电子邮件技术**，保证邮件从发出到接收的整个过程中，内容保密、无法修改且不可否认
- **传输安全增强技术**，在网络层或传输层使用安全协议（IPsec, SSL/TLS）来保证应用层的电子邮件在安全的传输通道上进行传输
- **邮件服务器安全增强**





内容提纲

1

电子邮件安全问题

2

安全电子邮件标准PGP

3

WebMail安全威胁及防范

4

垃圾邮件防范





安全电子邮件标准

- 端到端的安全电子邮件标准和协议主要有三种
 - PEM (Privacy Enhanced Mail, 隐私增强电子邮件):
 - S/MIME (Secure/Multipurpose Internet Mail Extensions, 安全/多用途因特网邮件扩展)
 - PGP (Pretty Good Privacy, 优良隐私保护)





PEM

- 由美国RSA实验室基于RSA和DES算法开发的安全电子邮件方案。它在电子邮件标准格式上增加了加密、认证、消息完整性保护和密钥管理功能。
 - 由于PEM在MIME之前提出的，因此它并不支持MIME，只支持文本信息。PEM依赖于PKI并遵循X.509认证协议，而当时要建立一个可用的PKI并不是一件容易的事



S/MIME

- S/MIME基于PEM，使用RSA提出的PKCS和MIME来增强Email的安全（对邮件主体进行**消息完整性保护、签名和加密**后作为附件发送）
 - S/MIME v1是1995年完成的（MIME是1992年推出的），v2在IETF的RFC2311和RFC2312中定义，v3在RFC 3850和RFC 3851中定义（这些RFC是信息文件，而不是标准或建议的标准）
 - S/MIME不仅用于实现安全电子邮件传输，任何支持MIME格式的数据传输机制或协议（如HTTP）均可用





PGP

- PGP由美国人菲利普·齐默尔曼于1991年开发出来的。PGP既是一个特定的安全电子邮件应用软件，也是一个安全电子邮件标准。
 - 1997年7月，PGP Inc.与齐默尔曼同意由IETF制定一项公开的互联网安全电子邮件标准，称作OpenPGP，RFC 2440, 3156, 4880, 5581, 6637等
 - OSF开发了“GnuPG”（GPG）：Gpg4win, KGPG, Seahorse, MacGPG, iPGMail, OpenKeychain等
 - OpenPGP联盟（<http://www.openpgp.org>）
-



PGP

- PGP最常用于安全电子邮件传输，但它也可以用于任何需要保证传输**机密性、完整性和认证**的应用中。
- 齐默曼开发PGP的故事



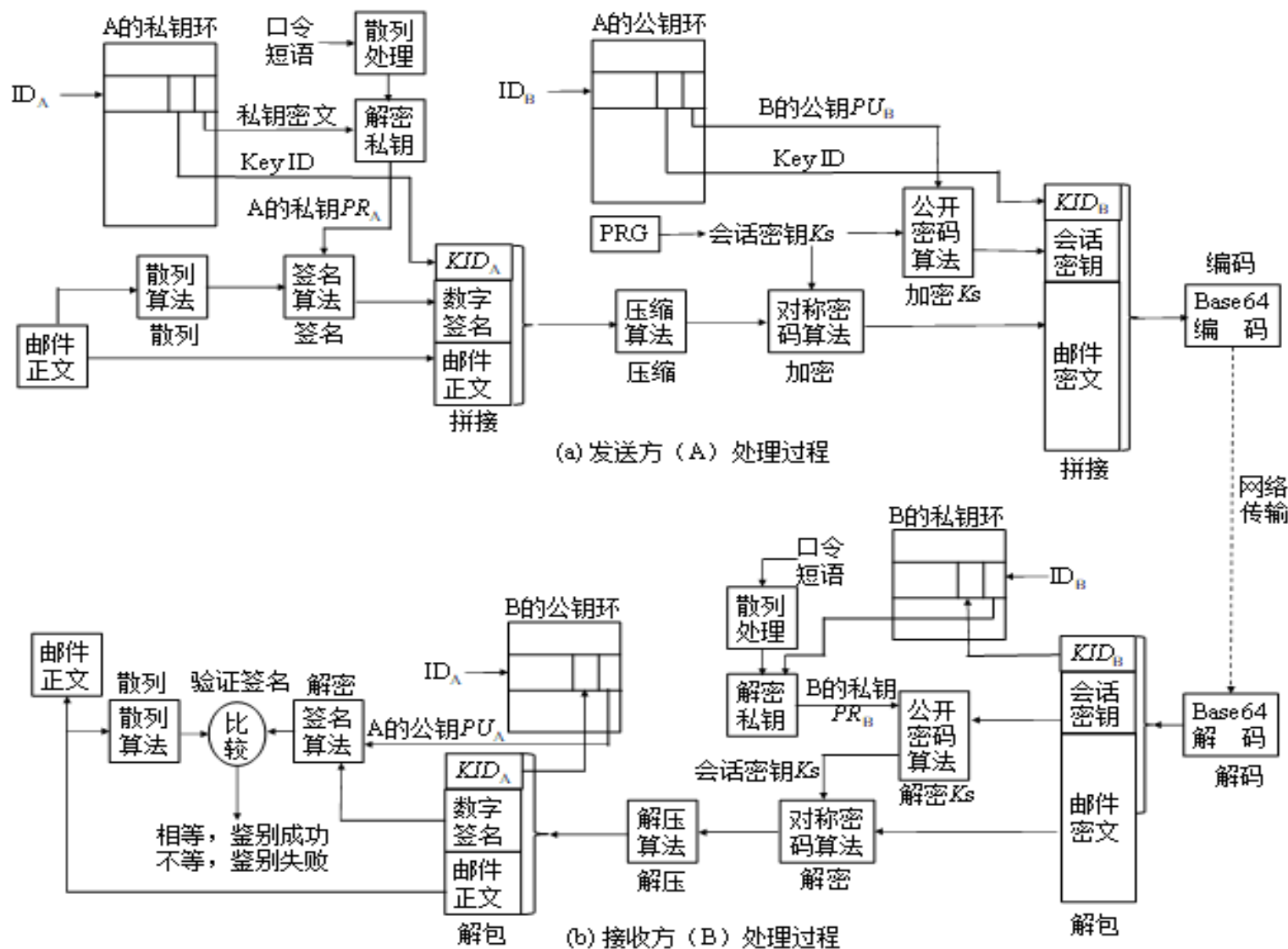


PGP功能

功能服务	采用的算法	说明
数字签名（包括身份鉴别）	散列算法：SHA-1, SHA224, SHA256, SHA384, SHA512, MD5, RIPEMD160等；签名算法：DSS 或 RSA	先用散列函数，如 SHA-1 产生消息的散列码，然后用 DSS 或 RSA 算法对散列码进行签名
消息加密	对称密码算法：CAST-128, IDEA, 3DES, AES 公开密码算法：RSA, Diffie-Hellman	消息用一次性会话密钥（对称密钥）加密，会话密钥用接收方的公钥加密
压缩	ZIP , ZLIB, BZIP2	消息用 ZIP / ZLIB / BZIP2 算法压缩后存储或传送
邮件兼容性	Radix 64	邮件应用安全透明，加密后的消息用 Radix 64 转换（也就是 MIME 的 Base64 编码）
数据分段		为了满足邮件的大小限制，支持分段和重组



PGP发送和接收邮件过程





PGP发送和接收邮件过程

- 讨论：签名、加密、压缩的顺序问题





PGP发送和接收邮件过程

- 讨论：兼容性考虑（Base64编码）
 - 为什么要进行Base64编码？
 - 对性能的影响如何？



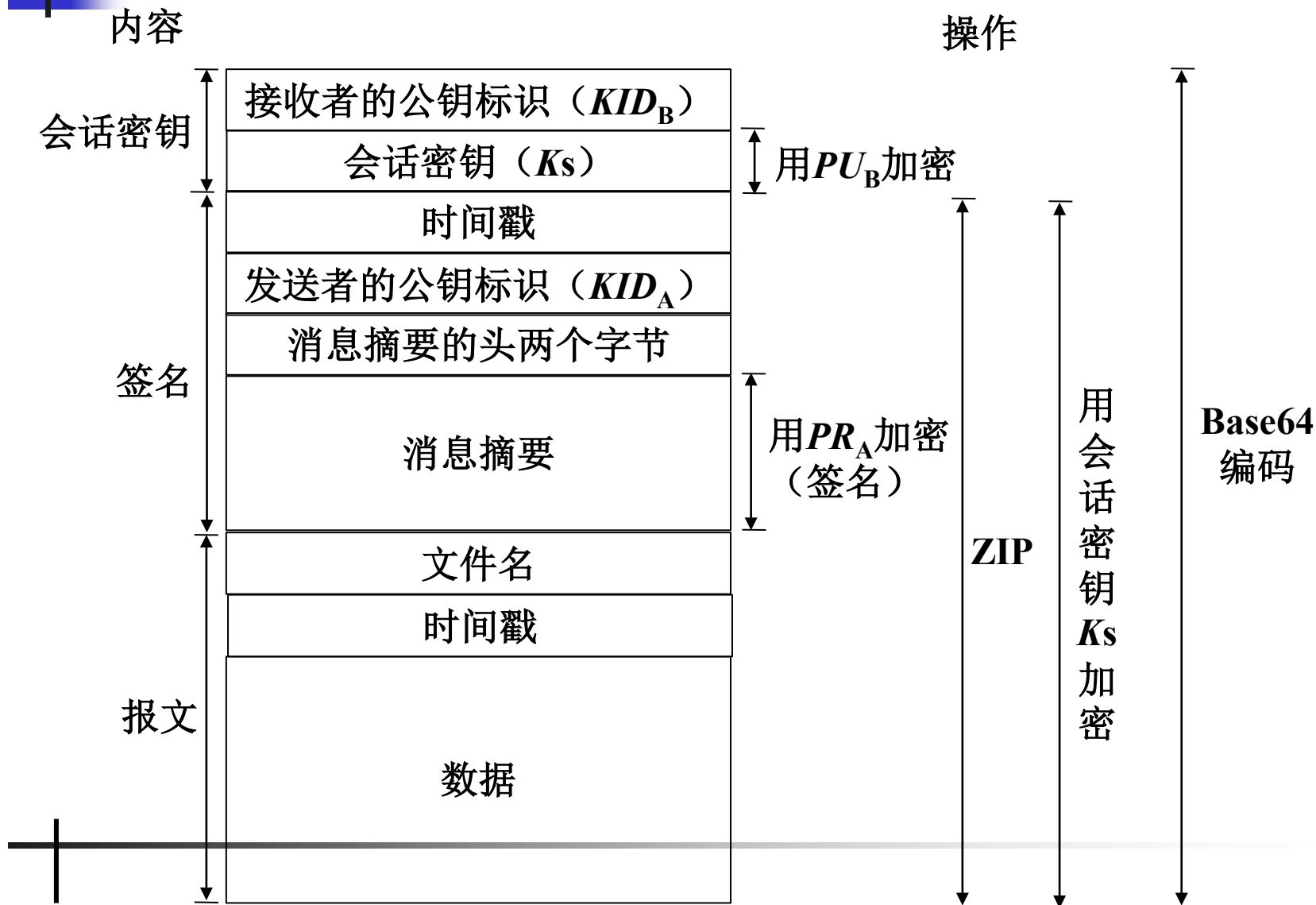


PGP发送和接收邮件过程

- 讨论：分段与重装
 - 为什么要分段？
 - 如果分段，会话密钥部分和签名部分在第几个报文段？



PGP消息格式





PGP密钥管理

- 会话密钥生成与管理

- 会话密钥 K_s 是由基于美国国家标准“金融机构密钥管理（大规模）”（ANSI X 9.17）中定义的随机数生成方法的伪随机数产生器（Pseudorandom number Generator, PRG）产生的128位随机数，只用于一条消息的一次加解密过程
- K_s 使用接收方的公钥加密后发送给接收方。这种会话密钥的安全交换方法称为“**数字信封**”





PGP密钥管理

- 公开密码算法密钥管理，用户A获取用户B的公钥主要方式包括：
 - 物理交付
 - 电话验证
 - 通过可信的第三方
 - 通过可信的认证中心（CA）





PGP密钥管理

- 公开密码算法密钥管理
 - 每个用户都有一个**公钥环**（Public Key Ring）和**私钥环**（Private Key Ring），均用表型数据结构存储
 - 公钥环存储该用户知道的其他用户的公钥，私钥环存储用户自己的所有公钥/私钥对。PGP允许一个用户同时拥有多个公钥/私钥对，主要目的有两个：一是经常变换密钥，以增强安全性；二是多个密钥对可以支持同一时刻与多个用户进行通信





PGP密钥管理

- 公开密码算法密钥管理
 - PGP采用**密钥标识符**（密钥ID，Key ID）来表示密钥，并建立密钥标识和对应公钥/私钥间的映射关系
 - 私钥存储方法：加密私钥的**口令短语**（passphrase）
p，加密后存储在私钥环中





PGP信任关系

- PGP信任模型：以用户为中心的信任模型（信任网模型，Web of Trust）
 - 没有一个统一的认证中心来管理用户公钥，每个人都可以作为一个CA对某个用户的公钥签名，以此来说明这个公钥是否有效（可信）
 - 当用户接收到新的公钥时，首先要检查公钥证书的签名者，然后根据这个签名者的信任程度计算出该公钥的合法性，如果合法才能把它插入到自己的公钥环中





PGP信任关系

- 公钥加密体制基础——用户必须确信他所拿到的公钥属于它看上去属于的那个人（公钥的真实性）

公钥介绍机制



用户要得到介绍人真实公钥并信任介绍人
（相对比较容易，而黑客想假冒很困难）





PGP信任关系

- 公钥环中每一个公钥项都有表示信任度的字段，包括：
 - 密钥合法性字段：合法和不合法
 - 签名可信性字段：不信任、部分信任、一直信任和绝对信任
 - 拥有者可信性字段：不信任、部分信任、一直信任和绝对信任，由用户自己指定。





PGP信任关系

- **签名可信性字段**的赋值方法：
 - 一个公钥可能有一个或多个签名证书，当为该公钥插入一个签名到公钥环中时，PGP首先搜索公钥环，查找签名者是否是已知公钥的拥有者。如果是，则将拥有者可信性字段中的标志赋给签名可信性字段；否则，将签名可信性字段赋值为不信任。





PGP信任关系

- **密钥合法性**字段的取值：
 - 由此公钥的所有签名的签名可信性字段的取值计算得到。计算方法：如果该公钥的签名可信性字段中至少有一个标志为绝对信任，则此公钥的密钥合法性字段标志为合法；否则，PGP计算所有签名信任值的加权和，即签名可信性字段标志为一直信任的权重为 $1/X$ ，标志为部分信任的权重为 $1/Y$ 。





PGP信任关系

- **拥有者可信性**字段取值：
 - 当用户A往公钥环中插入一个新的公钥时，PGP必须为该公钥的拥有者可信性字段设定一个标志。如果用户A插入的新公钥是自己的，则它也将被插入到用户A的私钥环中，PGP自动指定其密钥合法性字段标志为密钥合法，拥有者可信性字段标志为绝对信任；否则，PGP将询问用户A，让用户给定信任级别。





PGP安全性

- 口令或私钥的泄密
 - 口令泄露或被破解
 - 私钥文件的保护
- PGP缺少PKI体系那样严格的证书撤销机制，很难确保没有人使用一个已不安全的密钥，是PGP安全体系中比较薄弱的环节



PGP安全性

- PGP使用的公开密码算法、对称密码算法、安全

散列

SHA-1 is a Shambles

First Chosen-Prefix Collision on SHA-1
and Application to the PGP Web of Trust

Gaëtan Leurent¹ and Thomas Peyrin^{2,3}

¹ Inria, France

² Nanyang Technological University, Singapore

³ Temasek Laboratories, Singapore

gaetan.leurent@inria.fr, thomas.peyrin@ntu.edu.sg

<https://sha-mbles.github.io/>

omas

前缀冲突

SHA-1的

- GitHub 2.2.18 (2019.11.25发布)：对2019-01-

19之后基于SHA-1创建的身份签名视为无效

利用SHA-1冲突伪造证书

	Message A	Message B
0x0000	99 04 0d 04 7f e8 17 80 01 20 00 ff 4b 65 79 20 69 73 20 70 61 72 74 20 6f 66 20 61 20 63 6f 6c 6c 69 73 69 6f 6e 21 20 49 74 27 73 20 61 20 74 72 61 70 21 79 c6 1a f0 af cc 05 45 15 d9 27 4e	99 03 0d 04 7f e8 17 80 01 18 00 ff 50 72 61 63 74 69 63 61 6c 20 53 48 41 2d 31 20 63 68 6f 73 65 6e 2d 70 72 65 66 69 78 20 63 6f 6c 6c 69 73 69 6f 6e 21 1d 27 6c 6b a6 61 e1 04 0e 1f 7d 76
0x0008	73 07 62 4b 1d c7 fb 23 98 8b b8 de 8b 57 5d ba 7b 9e ab 31 c1 67 4b 6d 97 43 78 a8 27 73 2f f5 85 1c 76 a2 e6 07 72 b5 a4 7c e1 ea c4 0b b9 93 c1 2d 8c 70 e2 4a 4f 8d 5f cd ed c1 b3 2c 9c f1	7f 07 62 49 dd c7 fb 33 2c 8b b8 c2 b7 57 5d be c7 9e ab 2b e1 67 4b 7d b3 43 78 b4 cb 73 2f e1 89 1c 76 a0 26 07 72 a5 10 7c e1 f6 e8 0b b9 97 7d 2d 8c 68 52 4a 4f 9d 5f cd ed cd 0b 2c 9c e1
0x0010	9e 31 af 24 29 75 9d 42 e4 df db 31 71 9f 58 76 23 ee 55 29 39 b6 dc dc 45 9f ca 53 55 3b 70 f8 7e de 30 a2 47 ea 3a f6 c7 59 a2 f2 0b 32 0d 76 0d b6 4f f4 79 08 4f d3 cc b3 cd d4 83 62 d9 6a	92 31 af 26 e9 75 9d 52 50 df db 2d 4d 9f 58 72 9f ee 55 33 19 b6 dc cc 61 9f ca 4f b9 3b 70 ec 72 de 30 a0 87 ea 3a e6 73 59 a2 ee 27 32 0d 72 b1 b6 4f ec c9 08 4f c3 cc b3 cd d8 3b 62 d9 7a
0x0018	9c 43 06 17 ca ff 6c 36 c6 37 e5 3f de 28 41 7f 62 6f ec 54 ed 79 43 a4 6e 5f 57 30 f2 bb 38 fb 1d f6 e0 09 00 10 d0 0e 24 ad 78 bf 92 84 19 93 60 8e 8d 15 8a 78 9f 34 c4 6f e1 e6 02 7f 35 a4	90 43 06 15 0a ff 6c 26 72 37 e5 23 e2 28 41 7b de 6f ec 4e cd 79 43 b4 4a 5f 57 2c 1e bb 38 ef 11 f6 e0 0b c0 10 d0 1e 90 ad 78 a3 be 64 19 97 dc 8e 8d 0d 3a 78 9f 24 c4 6f e1 ea ba 7f 35 b4
0x0020	cb fb 82 70 76 c5 0e ca 0e 8b 7c ca 69 bb 2c 2b 79 02 59 f9 bf 95 70 dd 8d 44 37 a3 11 5f af f7 c3 ca c0 9a d2 52 66 05 5c 27 10 47 55 17 8e ae ff 82 5a 2c aa 2a cf b5 de 64 ce 76 41 dc 59 a5	c7 fb 82 72 b6 c5 0e da ba 8b 7c d6 55 bb 2c 2f c5 02 59 a3 9f 95 70 cd a9 44 37 bf fd 5f af e3 cf ca c0 98 12 52 66 15 e8 27 10 5b 79 17 8e aa 43 82 5a 34 1a 2a cf a5 de 64 ce 7a f9 dc 59 b5
0x0028	41 a9 fc 9c 75 67 56 e2 e2 3d c7 13 c8 c2 4c 97 90 aa 6b 0e 38 a7 f5 5f 14 45 2a 1c a2 85 0d dd 95 62 fd 9a 18 ad 42 49 6a a9 70 08 f7 46 72 f6 8e f4 61 eb 88 b0 99 33 d6 26 b4 f9 18 74 9c c0	4d a9 fc 9e b5 67 56 f2 56 3d c7 0f f4 c2 4c 93 2c aa 6b 14 18 a7 f5 4f 30 45 2a 00 4e 85 0d c9 99 62 fd 98 d8 ad 42 59 de a9 70 14 db 46 72 f2 32 f4 61 f3 38 b0 99 23 d6 26 b4 f5 a0 74 9c d0
0x0030	27 fd dd 6c 42 5f c4 21 68 35 d0 13 4d 15 28 5b ab 2c b7 84 a4 f7 cb b4 fb 51 4d 4b f0 f6 23 7c f0 0a 9e 9f 13 2b 9a 06 6e 6f d1 7f 6c 42 98 74 78 58 6f f6 51 af 96 74 7f b4 26 b9 87 2b 9a 88	2b fd dd 6e 82 5f c4 31 dc 35 d0 0f 71 15 28 5f 17 2c b7 9e 84 f7 cba 4 df 51 4d 57 1c f6 23 68 fc 0a 9e 9d 32 2b 9a 16 da 6f d1 63 40 42 98 70 c4 58 6f ee e1 af 96 64 7f b4 26 b5 3f 2b 9a 98
0x0038	e4 06 3f 59 bb 33 4c c0 06 50 f8 3a 80 c4 27 51 b7 19 74 d3 00 fc 28 19 a2 e8 f1 a3 2c 1b 51 cb 18 e6 bf c4 db 9b ae f6 75 d4 aa f5 b1 57 4a 04 7f 8f 6d d2 ec 15 3a 93 41 22 93 97 4d 92 8f 88	e8 06 3f 55 7b 33 4c d0 b2 50 f8 26 bc c4 27 55 0b 19 74 c9 20 fc 28 09 86 e8 f1 ff c0 1b 51 df 14 e6 bf c6 1b 9b ae e6 c1 d4 aa e9 9d 57 4a 00 c3 8f 6d ca 5c 15 3a 83 41 22 93 9b f5 92 8f 98
0x0040	ce d9 36 3c fe f9 7c e2 e7 42 bf 34 c9 6b 8e f3 87 56 76 fe a5 cc a8 e5 f7 de a0 ba b2 41 3d 4d e0 0e e7 1e e0 1f 16 2b db 6d 1e af d9 25 e6 ae ba ae 6a 35 4e f1 7c f2 05 a4 04 fb db 12 fc 45	c2 d9 36 3e 3e f9 7c f2 53 42 bf 28 f5 6b 8e f7 3b 56 76 e4 85 cc a8 f5 d3 de a0 a6 5e 41 3d 59 ec 0e e7 1c 20 1f 16 3b 6f 6d 1e b3 f5 25 e6 ae 06 ae 6a 2d fe f1 7c e2 05 a4 04 f7 63 12 fc 55
0x0048	4d 41 fd d9 5c f2 45 96 64 a2 ad 03 2d 1d a6 0a 73 26 40 75 d7 f1 e0 d6 c1 40 3a e7 a0 d8 61 df 3f e5 70 71 88 dd 5e 07 d1 58 9b 9f 8b 66 30 55	41 41 fd db 9c f2 45 86 d0 a2 ad 1f 11 1d a6 0e cf 26 40 6f f7 f1 e0 c6 e5 40 3a fb 4c d8 61 cb 33 e5 70 73 48 dd 5e 17 65 58 9b 83 a7 66 30 51

Figure 7: Chosen-prefix collision for SHA-1. The colors show the prefix, the birthday bits, and the near-collision blocks. Both messages have the same SHA-1: 8ac60ba76f1999a1ab70223f225aefdc78d4ddc0



利用SHA-1冲突伪造证书

```
messageA(sha-mbles.github.io/messageA)
```

```
messageB(sha-mbles.github.io/messageB)
```

```
sha1sum验证:
```

```
sha1sum messageA && sha1sum messageB
```

```
8ac60ba76f1999a1ab70223f225aefdc78d4ddc0 messageA
```

```
8ac60ba76f1999a1ab70223f225aefdc78d4ddc0 messageB
```



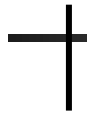


利用SHA-1冲突伪造证书

6.1 Exploiting a Chosen-prefix Collision

We now focus on the identity certificates that will be hashed and signed. Following RFC 4880 [CDF⁺07], the hash function receives the public key packet, then a UserID or user attribute packet, and finally a signature packet and a trailer. The idea of the attack is to build two public keys of different sizes, so that the remaining fields to be signed are misaligned, and we can hide the UserID of key A in another field of key B. Following RFC 4880, the signature packet is protected by a length value at the beginning *and at the end*, so that we have to use the same signature packet in key A and key B (we cannot stuff data in the hashed subpacket). Therefore, we can only play with the UserID and/or user attribute packets. Still, a user attribute packet with a JPEG image gives us enough freedom to build colliding certificates, because typical JPEG readers ignore any bytes after the End of Image marker (ff d9). This gives us some freedom to stuff arbitrary data in the certificate.

More precisely, we build keys A and B as follows. Key A contains a 8192-bit RSA public key, and a UserID field corresponding to Alice. On the other hand, key B contains a 6144-bit RSA public key, the UserID of Bob and a JPEG image. Therefore, when Bob gets a certification signature of his key, the signer will sign two certificates: one containing his public key and UserID, and another one containing the public key and the image. The public keys A and B and the image are crafted in such a way to generate a collision between the certificates with the key A and Alice's UserID, and the certificate with key B and the image.



利用SHA-1冲突伪造证书

Figure 8 shows a template of the values included in the identity certificate: those values are hashed when signing a key, and we want the two hashes to collide. In this example, the UserID field of key A contains “Alice <alice@example.com>”, and the image in key B is a valid JPEG image that will be padded with junk data after the End of Image marker. The real JPEG file is 181 bytes long⁷ (from ff d8 to ff d9), and it is padded with 81 bytes, so that the file included in the key is 262 bytes long (here the padding includes 46 bytes corresponding to the end of the modulus of key A, 5 bytes corresponding to the exponent of key A, and 30 bytes corresponding to Alice’s UserID).

In Figure 8, we use the following symbols:

- 01 Bytes with a fixed value are fixed by the specifications, or chosen in advance by the attacker (length of fields, UserID, user attribute, ...)
- ?? Represent bytes that are determined by the chosen-prefix collision algorithm (the messages M and M' to generate a collision)
- !! Represent bytes that are selected after finding the collision, to generate an RSA modulus with known prime factors
- .. Represent bytes that are copied from the other certificate
- ** Represent time-stamps chosen by the attacker
- \$\$ Represent the time-stamp chosen by the signer

Underlined values correspond to packet headers (type and length).

利用SHA-1冲突伪造证书

To carry out the attack, we have to perform the following steps:

1. Build a chosen-prefix collision with prefixes “99 04 0d 04 ** ** ** ** 01 20 00” and “99 03 0d 04 ** ** **** 01 18 00”, after filling the ** with two arbitrary time-stamps. The chosen-prefix collision must have at most 10 near-collision blocks. This determines the ?? bytes of the keys.
2. Choose a tiny JPEG image to include in key B (fixed orange bytes), and an arbitrary UserID to include in key A (fixed yellow bytes)
3. Select the “!!” bytes in key B to make a valid modulus
4. Select the “!!” bytes in key A to make a valid modulus
5. Generate key B with the modulus and the padded JPEG. Ask for a signature of the key.
6. Copy the signature to key A.

We point out that the chosen-prefix collision is computed *before* choosing the UserIDs and images that will be used in the attack. Therefore, a single CPC can be reused to attack many different victims. This contrasts with attacks on X.509 certificates [SLdW07, SSA⁺09], where the identifier is hashed before the public key.

利用SHA-1冲突伪造证书

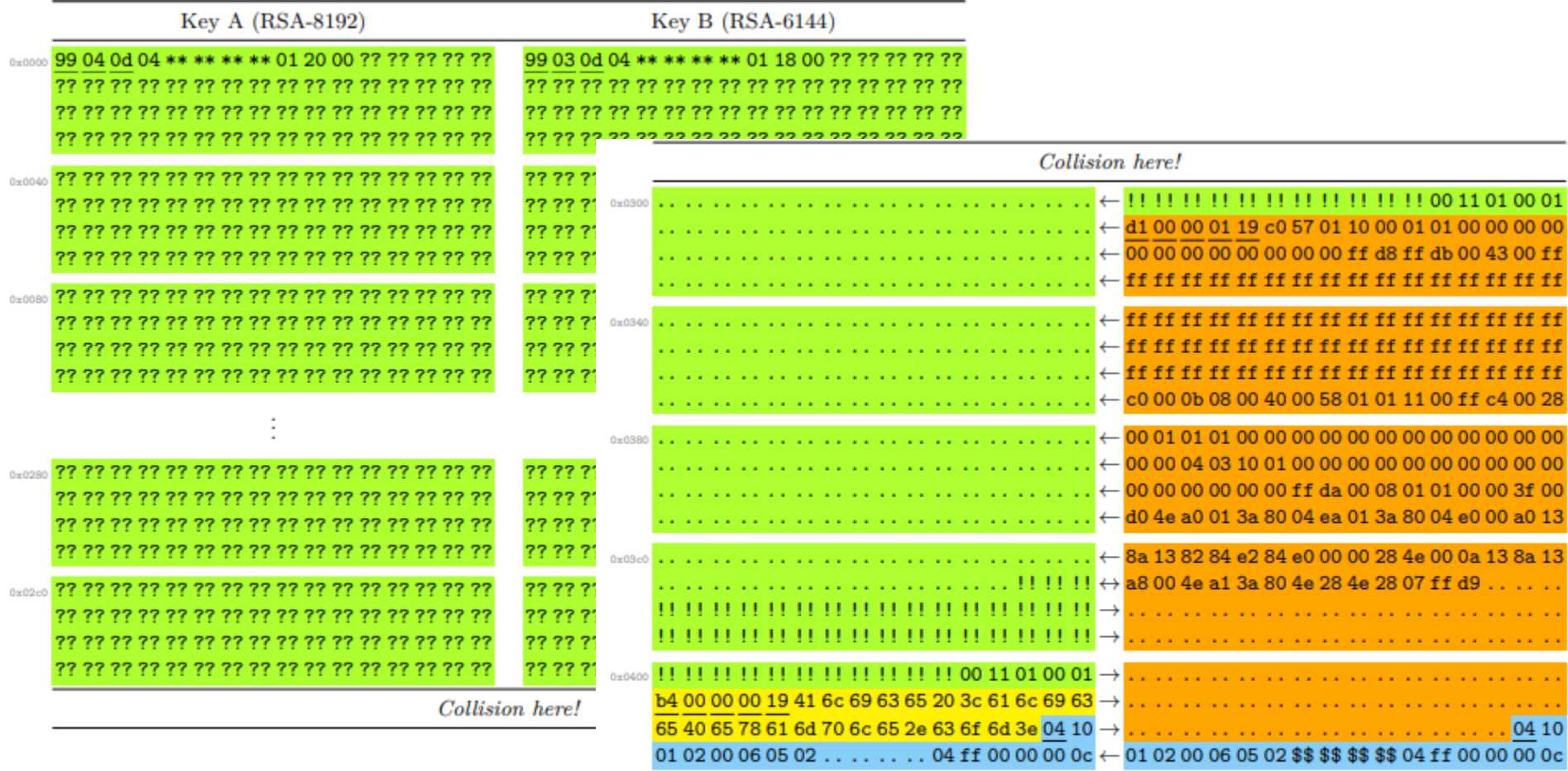


Figure 8: Construction of colliding OpenPGP identity certificates.

The colour corresponds to the packets hashed when computing the signature: first, the public key packet (with header), then the UserID or user attribute, and finally the signature packet and trailer. Arrows show when a value is chosen in one key and copied to the other.



利用SHA-1冲突伪造证书

已选择前缀来构建两个具有冲突SHA-1认证签名的PGP公钥。可以在下载以下两个具有不同用户名的示例密钥，并使用 `pgpdump -i` 检查它们，以查看由 `0xAFBB1FED6951A956` 发出的SHA-1签名是否相同：

`alice.asc(sha-mbles.github.io/alice.asc)`

`bob.asc(sha-mbles.github.io/bob.asc)`

为了避免恶意使用，密钥的创建日期很远。如果要使用pgp分析它们，则可以使用 `--ignore-time-conflict --ignore-valid-from` 选项，可以给命令前添加 `false-time @2145920400` 作为前缀）。





讨论

- 现有的加密电子邮件解决方案，如PEM，S/MIME, PGP等，大都是对邮件正文进行安全处理。为什么？





内容提纲

1

电子邮件安全问题

2

安全电子邮件标准**PGP**

3

WebMail安全威胁及防范

4

垃圾邮件防范



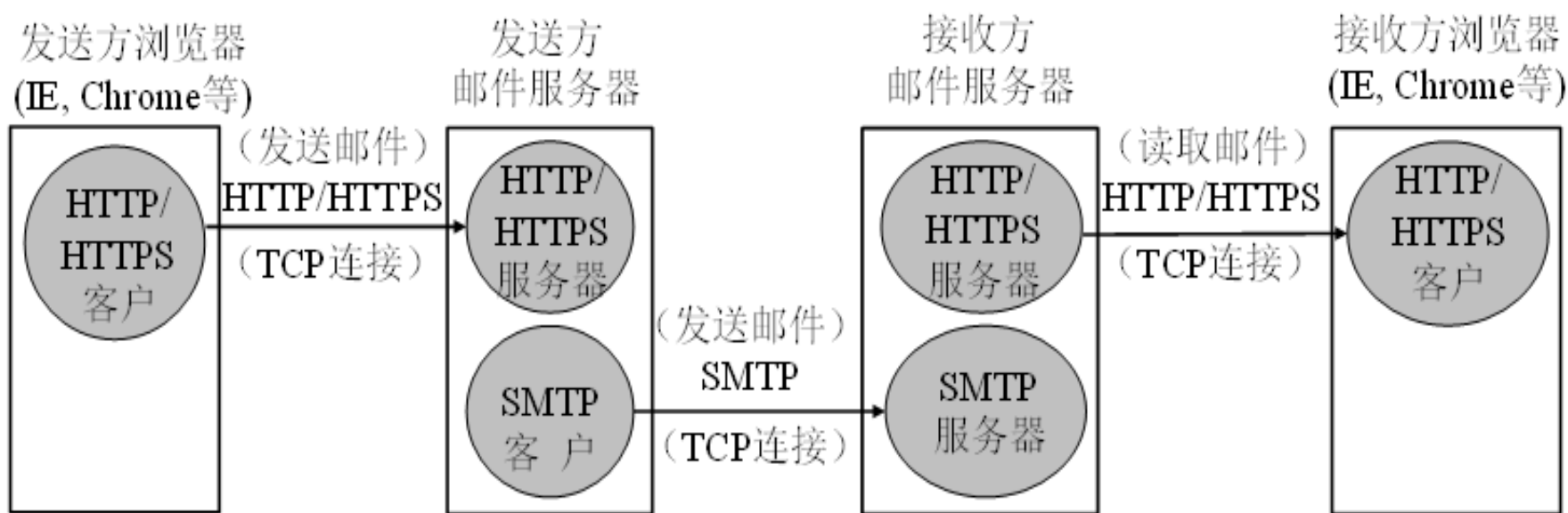


WebMail

- WebMail不需借助专门的邮件客户端，只要能
用浏览器上网就能收发邮件，极大地方便了用
户。但是，WebMail的使用也带来的新的安全
威胁，前面介绍的Web应用所面临的很多安全
问题同样在WebMail中存在



WebMail



(b) WebMail收发邮件



WebMail安全问题

- WebMail暴力破解

- 防范：禁用账户、禁止IP地址、登录检验

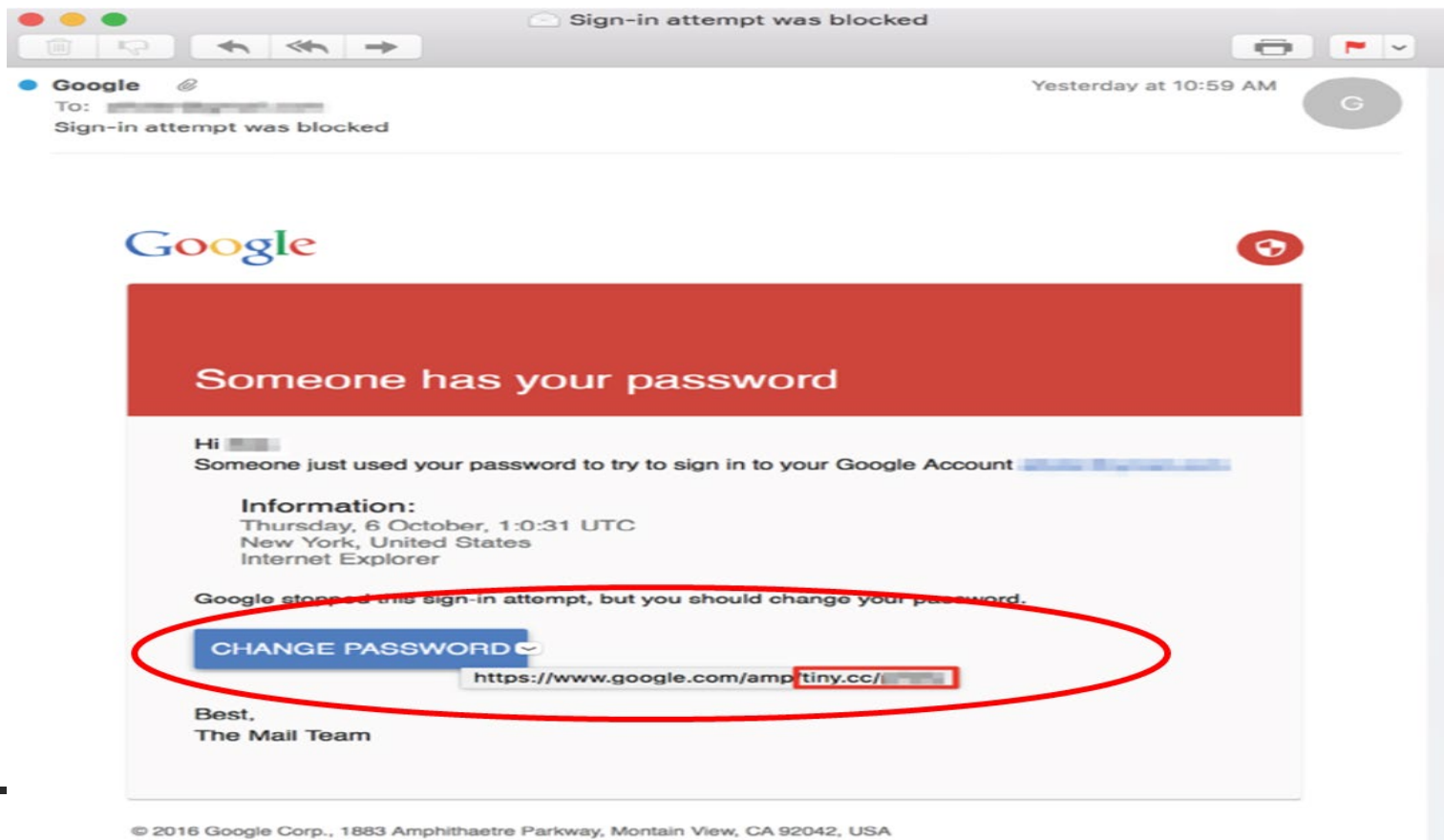
- 恶意HTML邮件

- 利用HTML邮件，攻击者能进行电子邮件欺骗，甚至欺骗用户更改自己的邮箱密码
 - 在HTML邮件中嵌入恶性脚本程序，攻击者还能进行很多破坏攻击，如修改注册表、非法操作文件、格式化硬盘、耗尽系统资源、修改“开始”菜单等，甚至能删除和发送用户的邮件、访问用户的地址簿、修改邮箱帐户密码等



恶意邮件示例

- 2016年，希拉里竞选团队主席收到的伪装成Google的警告邮件的钓鱼邮件



恶意邮件示例

- 假冒网易邮箱管理员的身份给用户发送的安全告警邮件



<http://hostingdan.info/mailservervip.163.com.php>

恶意邮件示例

■ 假冒网易邮箱管理员的身份给用户发送的安全告警邮件

The screenshot displays a NetEase email client interface. On the left is a sidebar with navigation options: 收信 (Receive), 写信 (Compose), 收件箱 (Inbox), 红旗邮件 (Red Flag Mail), 待办邮件 (To Do Mail), 星标联系人邮件 (Starred Contact Mail), 草稿箱 (Drafts), 已发送 (Sent), 订阅邮件 (8) (Subscribed Mail), 其他4个文件夹 (Other 4 folders), 邮件标签 (Mail Labels), 邮箱中心 (Mailbox Center), 文件中心 (File Center), and 邮箱附件 (Mailbox Attachments). The main content area shows an email header with the subject '紧急通知28775' (Urgent Notice 28775), sender 'VIP用户警告-secure-accessMail<qualitym@shxgroup.com>', and recipient '我<[redacted]@vip.163.com>'. The email body contains a warning message from '网易 NETEASE' (NetEase) stating that the user's mailbox is nearly full (799.21M / 1000.00M) and that they are waiting for mail. It warns that the account will be blocked if the capacity is not increased within 24 hours to 25GB, and that all data will be lost. A button labeled '在这里管理您的容量' (Manage your capacity here) is highlighted with a red box. At the bottom of the email body, a copyright notice '© 1997 - 2019' is visible.

收信 写信

<< 返回 回复 回复全部 转发 删除 举报 标记为 移动到 更多

收件箱

红旗邮件

待办邮件

星标联系人邮件

草稿箱

已发送

订阅邮件 (8)

> 其他4个文件夹

> 邮件标签

> 邮箱中心

文件中心

邮箱附件

紧急通知28775

发件人: VIP用户警告-secure-accessMail<qualitym@shxgroup.com>

收件人: 我<[redacted]@vip.163.com>

时间: 2020年07月14日 01:50 (星期二)

警告!

我的邮箱:

邮箱使用: 799.21M / 1000.00M 您正在等待邮件....

您的邮箱几乎达到最大容量, 您的帐户将很快被阻止。为避免这种浪费, 请在24小时内更新到25GB。它是免费升级。确保您完成此任务以避免自动暂停邮件, 所有数据电子邮件将丢失。

在这里管理您的容量

© 1997 - 2019

[http://count.mail-163-comhunw.com/js5?user=\[redacted\]&dom=vip&text=163.com](http://count.mail-163-comhunw.com/js5?user=[redacted]&dom=vip&text=163.com)

恶意邮件示例

VIP 尊贵邮
VIP.163.com

@vip.163.com

9+

设置

个人中心

VIP客服

续费

English

退出

首页

通讯录

邮箱应用

收件箱

重要通知 ×

收信

写信

<< 返回

回复

回复全部

转发

删除

举报

标记为

移动到

更多

收件箱 (1)

红旗邮件

待办邮件

星标联系人邮件

草稿箱

已发送

订阅邮件

> 其他4个文件夹

> 邮件标签

> 邮箱中心

文件中心

邮箱附件

尊敬的网易VIP邮箱用户:

我们的记录表明,最近有 (2) 条消息已发送给您,但尚未成功传递到您的邮箱。我们建议您及时还原消息,以避免被拒绝。
请给我们24小时以恢复消息,或者

在此处登录以恢复消息

无法还原邮件将导致您的邮件被退回

谢谢。

网易公司版权所有©1997-2020

今日优选

http://count.mail.163.com.neteaseushwmail.com/gs85872188/nkzmljbr63.php

我的视频

头条推荐

热点资讯



WebMail安全问题

■ Cookie会话攻击

- 攻击者获取用户WebMail的Cookie信息后，就能很容易地侵入用户的WebMail。攻击者获取用户WebMail的Cookie信息的方法主要有内网监听和XSS攻击
- 含有恶性脚本程序的HTML邮件能使攻击者获取WebMail的Cookie信息
- WebMail系统应该避免使用持久型Cookie会话跟踪，使攻击者在Cookie会话攻击上不能轻易得逞





内容提纲

1

电子邮件安全问题

2

安全电子邮件标准**PGP**

3

WebMail安全威胁及防范

4

垃圾邮件防范





垃圾邮件（Spam）

- 从用户的角度看，正常邮件与垃圾邮件的主要区别是该邮件是否是用户所希望收到的邮件。用户查看邮件内容后，很容易判断出一封邮件是不是自己想要的邮件。但是，邮件服务器要判断一封邮件是不是垃圾邮件则要困难得多





垃圾邮件特征

- 基本特征如下：

- 内容的重复性：内容的大量重复
- 信息的合法性：查不到发件人信息
- 时间的有效性：不正常的发送时间
- 地址的有效性：无效的发送源地址
- 邮箱名的有效性：名称由无意义字符串组成
- HTML的合法性：不合法的HTML tag，大量图片
- 发送行为特征：大批量、时间短、发送源变换





反垃圾邮件

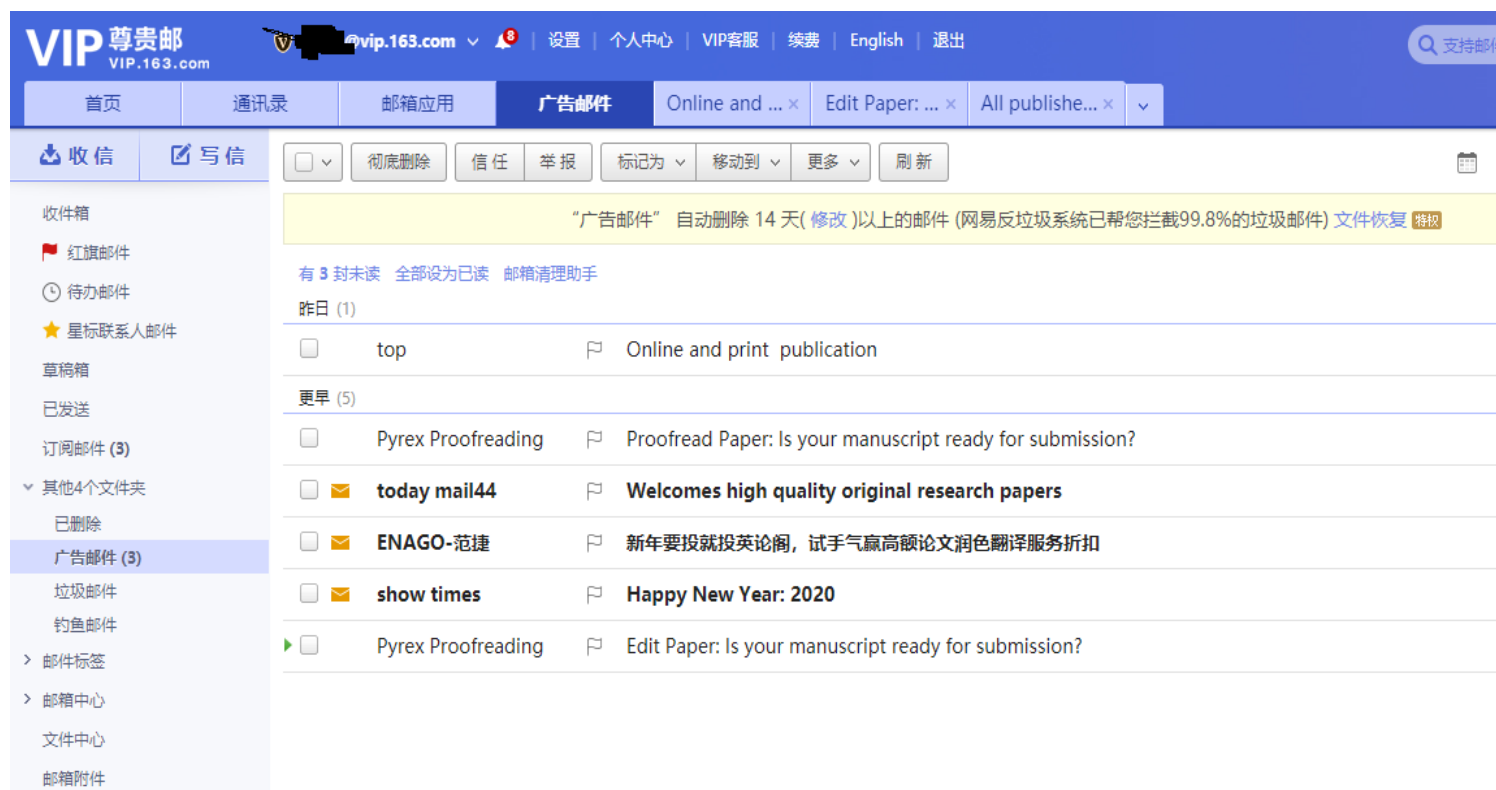
- 反垃圾邮件方法

- 基于地址的垃圾邮件检测：黑白名单检测、反向域名验证技术
- 基于内容的垃圾邮件检测：分布式协作法、规则过滤法、统计过滤法、关键词过滤法
- 基于行为的垃圾邮件检测：基于邮件通信拓扑相似性的垃圾邮件检测、基于邮件用户社交关系的垃圾邮件检测、基于SMTP连接行为的垃圾邮件



反垃圾邮件

■ 网易邮箱的反垃圾邮件系统



The screenshot displays the NetEase Mailbox (VIP.163.com) interface. The top navigation bar includes links for '首页' (Home), '通讯录' (Address Book), '邮箱应用' (Mailbox Apps), '广告邮件' (Advertisement Emails), and several open tabs. The left sidebar shows the folder structure, with '广告邮件 (3)' (Advertisement Emails (3)) selected. The main content area shows a list of emails in the 'Spam' folder, with a yellow banner indicating that 99.8% of spam emails have been blocked by the system.

“广告邮件” 自动删除 14 天(修改)以上的邮件 (网易反垃圾系统已帮您拦截99.8%的垃圾邮件) 文件恢复 特快

有 3 封未读 全部设为已读 邮箱清理助手

昨日 (1)

<input type="checkbox"/>	top	Online and print publication
更早 (5)		
<input type="checkbox"/>	Pyrex Proofreading	Proofread Paper: Is your manuscript ready for submission?
<input type="checkbox"/>	today mail44	Welcomes high quality original research papers
<input type="checkbox"/>	ENAGO-范捷	新年要投就投英论阁, 试手气赢高额论文润色翻译服务折扣
<input type="checkbox"/>	show times	Happy New Year: 2020
<input type="checkbox"/>	Pyrex Proofreading	Edit Paper: Is your manuscript ready for submission?



本章小结





作业

