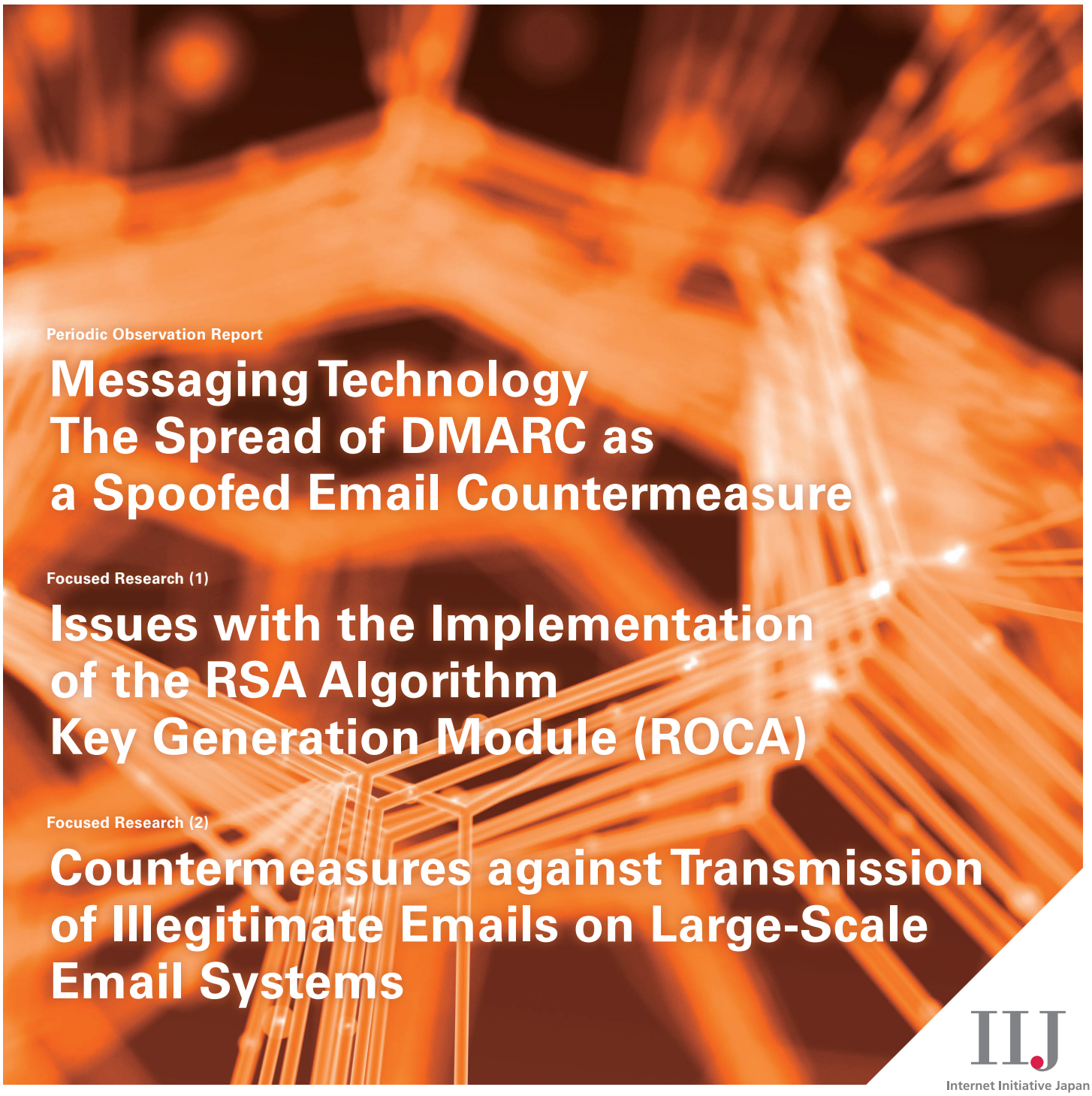
IIR

Internet
Infrastructure
Review

Jun.2018
Vol.39



Periodic Observation Report

Messaging Technology The Spread of DMARC as a Spoofed Email Countermeasure

Focused Research (1)

Issues with the Implementation of the RSA Algorithm Key Generation Module (ROCA)

Focused Research (2)

Countermeasures against Transmission of Illegitimate Emails on Large-Scale Email Systems

IIJ
Internet Initiative Japan

Internet Infrastructure Review

June 2018 Vol.39

Executive Summary

The Constitution of Japan guarantees the secrecy of communications, so telecommunications carriers such as IIJ, and their employees, are forbidden from encroaching upon the secrecy of communications they handle by the Telecommunications Business Act. That said, in carrying out a telecommunications business, we infringe upon the secrecy of communications by referring to communication logs for billing purposes, and header information to deliver packets to their destination. As an ISP, some of the activities such as spam filtering, OP25B, and the blocking of child pornography that we perform on a daily basis to provide peace of mind when using the Internet also interfere with the secrecy of communications. For this reason, careful arrangements are made to provide justifiable cause for noncompliance with the law, such as by obtaining customer consent.

Given this background, it was a big surprise to ISPs such as IIJ, and lawyers and consumer organizations also expressed strong concern, when emergency measures against Internet-based piracy sites compiled by the Japanese government's Intellectual Property Strategy Headquarters in April suggested it would be appropriate for ISPs to block particularly malicious piracy sites as an urgent stopgap measure until judicial measures could be put in place. Going forward, the government plans to set up a task force to discuss this, so we will be keeping a close eye on how this matter develops.

IIJ aims to introduce the wide range of technology that it researches and develops in this IIR, which comprises periodic observation reports that provide an outline of various data we obtain through the daily operation of services, as well as focused research where we examine specific areas of technology. In this volume, Chapter 1 is our periodic observation report, while our first focused research report in Chapter 2 discusses the ROCA vulnerability in a key generation module for the RSA cryptographic algorithm. In Chapter 3, our second focused research report covers email issues.

In the first half of Chapter 1, we examine trends in incoming spam detected on IIJ's email services for the period of 500 weeks from 2008 to 2017, and also discuss notable changes in 2017. In the latter half, we report on trends in the adoption and standardization of DMARC sender authentication technology, an effective anti-spam measure. We also take a look at key legal matters to consider when implementing it in Japan.

Chapter 2 is our first focused research report, where we examine a flaw called ROCA in the implementation of a key generation module for the RSA cryptographic algorithm that leads to encryption not providing the expected level of security. We also analyze the cause of the vulnerability, and consider the future impact of these research results.

In Chapter 3, we discuss the construction of large-scale email systems that IIJ offers to service providers with millions of users, and look at the countermeasures against improper use that we have accumulated over the course of operating these services. Most of the work performed by email system administrators involves correspondence caused by improper use of the email system, as well as associated troubleshooting. For this reason, implementing countermeasures against improper use in email systems contributes greatly to reducing the operational and management workload and providing stable services to end users.

IIJ continues to strive towards improving and developing its services daily, while maintaining the stability of the ICT environment. We will continue to provide a variety of services and solutions that our customers can take full advantage of as infrastructure that supports their business.

Junichi Shimagami
Mr. Shimagami is a Senior Executive Officer and the CTO of IIJ. His interest in the Internet led to him joining IIJ in September 1996. After engaging in the design and construction of the A-Bone Asia region network spearheaded by IIJ, as well as IIJ's backbone network, he was put in charge of IIJ network services. Since 2015, he has been responsible for network, cloud, and security technology across the board as CTO. In April 2017, he became chairman of the Telecom Services Association of Japan MVNO Council.

Messaging Technology
The Spread of DMARC as a Spoofed Email Countermeasure

1.1 Introduction

Here we report on trends in email with a focus on spam, as well as technological trends in anti-spam measures.

Since our first volume in 2008, we have reported on trends in the ratio of spam detected in incoming email on IIJ's email services as an indicator of changes in the volume of spam. Because of the renewal of the email system this fiscal year, however, this will be our last report in the current format. Going forward, we will report in a new format whenever there are major changes or shifts.

Regarding trends in technology, we continue to discuss sender authentication technology and report on the status of its adoption. We also give an overview regarding the implementation of DMARC sender authentication technology, for which certain legal matters were sorted out last year.

1.2 Spam Trends

In this section we report on changes in the ratios of spam detected by the spam filter provided through IIJ's email services, as an indicator of spam trends. This time we examine the results of all previous surveys, covering the period from Week 23 of 2008 (the week starting June 2, 2008) to Week 52 of 2017 (the week starting December 25, 2017), which covers exactly 500 weeks (Figure 1).

The average ratio of spam for 2017 was 30.5%. The average for 2016 was 39.9%, so the 2017 figure represents a decrease of 9.4 percentage points, but in 2015 the average was 24.7%, so the ratio is not simply falling. In fact, spam continues to include many phishing emails that spoof prominent companies, and malicious spam that leads to the execution of ransomware seems to be on the rise recently.

1.2.1 Phishing Email Authentication Results

In the previous report (Vol.35), we discussed the results of using sender authentication on spam disguised to appear as though it came from Microsoft. Since then, we have continued to observe phishing emails purporting to be from high-profile companies.

The Council of Anti-Phishing Japan publishes information*1 on phishing emails, so it is important to check whether incoming email is one of the phishing emails in recent circulation before opening URLs or HTML files within. However, the use of sender authentication technology offers an easier way to detect phishing emails. Many major companies already have implemented DMARC, enabling DMARC authentication to be used to detect the spoofing of sender information.

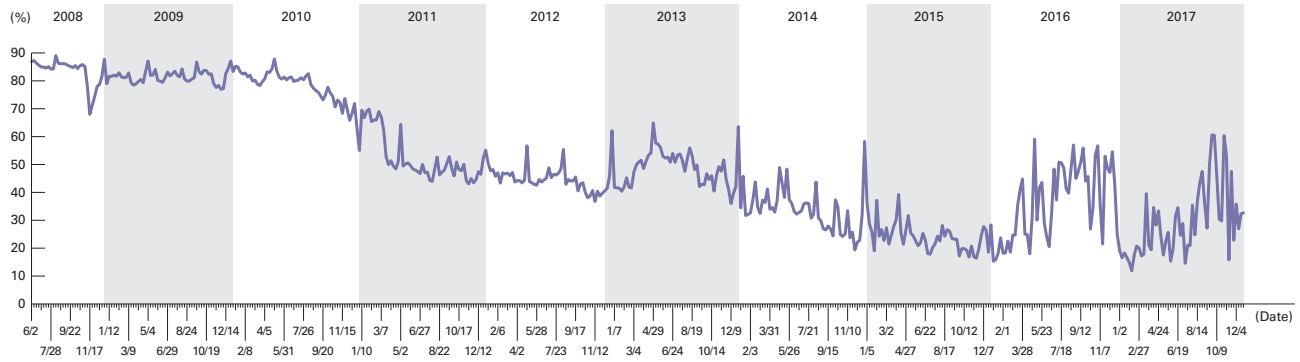


Figure 1: Spam Ratio Trends

*1 Council of Anti-Phishing Japan: Phishing-Related News (http://www.antiphishing.jp/news/alert/) (in Japanese).

Taking Apple as an example, emails supporting SPF, DKIM, and DMARC are sent for emails related to iCloud login. Phishing emails that have been sent recently use the same email.apple.com domain name as the genuine one in the sender information (RFC5322. From) email header. Naturally, SPF fails (softfail) because the sender is different, and DMARC authentication fails (fail) due to the lack of a DKIM signature (none). Moreover, because the DMARC record policy for email.apple.com is set to reject (p=reject), emails like this will not be delivered when determining whether to receive email in accordance with DMARC specifications.

Many emails purporting to be from Rakuten Ichiba or Rakuten Card are also still being sent. Similarly, these use the rakuten.co.jp or mail.rakuten-card.co.jp domain names in the sender information header, but a DMARC record has been configured for both domain names. As a result, DMARC authentication fails in each case, so it is easy to detect spoofed email. As the implementation of DMARC sender authentication progresses, it will be possible to eliminate this kind of unwanted email. Also, when managing domain names that are easily spoofed, it is best to configure DMARC records as a countermeasure for spoofing.

1.2.2 New Email Threats

The IC3 (Internet Crime Complaint Center) of the FBI in the United States published an Internet Crime Report for 2017*2. The topics covered in 2017 included BEC*3 and ransomware.

BEC is a type of fraud where email recipients are tricked into sending money using sophisticated techniques. It was reported that the IC3 received 15,690 complaints in 2017, representing a loss of over 675 million dollars.

Figure 2: Email Spoofing Apple

*2 FBI, “Latest Internet Crime Report Released” (https://www.fbi.gov/news/stories/2017-internet-crime-report-released-050718).
*3 BEC: Business Email Compromise.

In Japan, a major airline company was also tricked by an email misrepresented as being from a business partner requesting a change of remittance account in September 2017, and they made headlines when they announced in December of the same year that this resulted in damages of over 300 million yen. Of course, most email users may think they would never be fooled by such fraudulent email. But the fact is that many have fallen victim, and according to reports these crimes are prepared meticulously using very sophisticated techniques. To avoid meeting a similar fate, it is first necessary to implement robust technological countermeasures.

The WannaCry ransomware that was big news in Japan and other countries in May 2017 is a type of malicious program (malware) in a broad sense of the term. When a ransomware infection occurs, important files are encrypted and demands are made for payment by virtual currency or other means to obtain the decryption key. There are a variety of infection vectors, but one of these includes targeted attacks, so it is crucial to implement email countermeasures as well. Unlike conventional malware business models (obtaining confidential information to sell separately on the black market etc.), this method involves new techniques where money is obtained directly from the victim by demanding payment in virtual currency, preventing the recipient from being traced.

The IC3 reports that 1,783 complaints were identified as ransomware in 2017, causing over 2.3 million dollars in damages. This year, ransomware attacks in the U.S. city of Atlanta in March also caused extensive damages*4.

1.3 Trends in Email Technologies

Here we report on trends in the adoption and standardization of DMARC sender authentication and other related technologies. We also examine the legal handling that is important when implementing it in Japan.

1.3.1 DMARC Penetration

DMARC authentication is implemented for email received on IIJ email services. Figure 3 shows trends in the monthly average DMARC authentication results up until April 2018.

In the latest incoming email survey results for April 2018, the ratio of all email for which DMARC authentication was possible climbed to 19.3%, a record high among surveys to date. The ratio for pass authentication results was also the highest ever, at 12.2%. DMARC is still not at a level where it could be called widespread, but the number of domain names implementing it is growing at a steady rate.

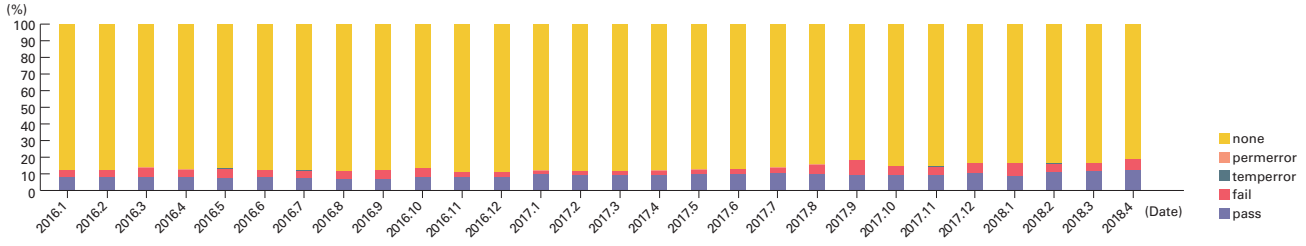


Figure 3: DMARC Authentication Result Trends

*4 The City of Atlanta, "Ransomware Cyberattack Information" (https://www.atlantaga.gov/government/ransomware-cyberattack-information).

Next, Figure 4 shows sender authentication result combinations for April 2018, including SPF and DKIM. For example, the DMARC+SPF+DKIM data category (8.8%) indicates the ratio of email that produced a pass result for DMARC, SPF, and DKIM collectively. In other words, we found that the most prevalent combination for domain names that could be authenticated with DMARC was domains that implemented both SPF and DKIM. This is the same combination as in the previous survey (Vol.35). The authentication with the highest ratio was SPF (35.4%). Including combinations with other authentications, the total pass rate for SPF is 69.3%, and we surmise that ease of implementation is a factor in its popularity. In the latest March 2018 report data*5 from the Ministry of Internal Affairs and Communications, the pass rate was over 90%.

Conversely, items marked "!(...)" indicate the ratio for which the authentication technology combinations listed in the brackets failed, and did not produce a single pass result. From Figure 4, we can see that "!(SPF)" by itself had the highest authentication failure rate at 6.5%. Although it is possible that the sender domain name is being spoofed, we believe that a considerable portion is email received after being forwarded, which is a case that SPF cannot authenticate properly.

Next, Figure 5 shows the ratio of domain names that could be authenticated using DMARC by TLD (Top-Level Domain). The TLD that succeeded most frequently were the ".com" domains (58.4%). Next were the ".jp" domains (7.0%), which produced results starkly different to the ".com" domains. Considering that this is incoming email in Japan, you could say the ratios are high for Australia*6 (".au," 4th, 2.8%) and the United Kingdom*7 (".uk," 6th, 2.1%), where implementation is being encouraged at the government agency level.

Based on volume, com shifts to 53.6% and jp to 43.4%, and these two TLDs account for the majority of DMARC domain names.

I mentioned that the implementation of DMARC is being encouraged at government agencies in Australia and the United Kingdom. The Department of Homeland Security (DHS) in the United States has also decided to enhance email and Web security at federal agencies (BOD 18-01)*8. This decision requires that DMARC records be configured with at minimum a "p=none" policy within 90 days. Also, a "p=reject" policy must be configured within a year. As already reported, if a "p=reject" and DMARC record policy is configured, it is more likely that the email recipient will reject messages when DMARC authentication fails. In short, to declare "p=reject", the email system, including SPF and DKIM settings, must be properly managed to prevent legitimate email from failing DMARC authentication. In this sense, it could be said that the U.S. DHS made a very important decision. We hope that the government and municipalities of Japan also look into doing this.

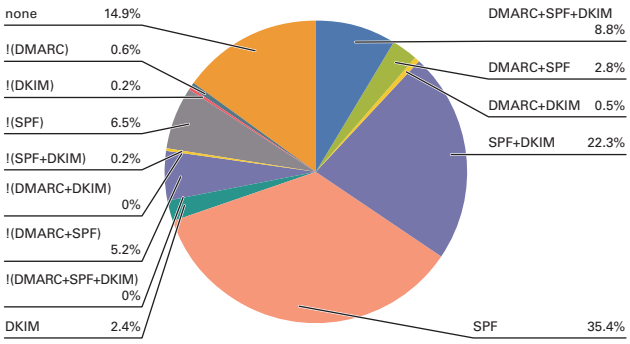


Figure 4: Sender Authentication Result Combinations (April 2018)

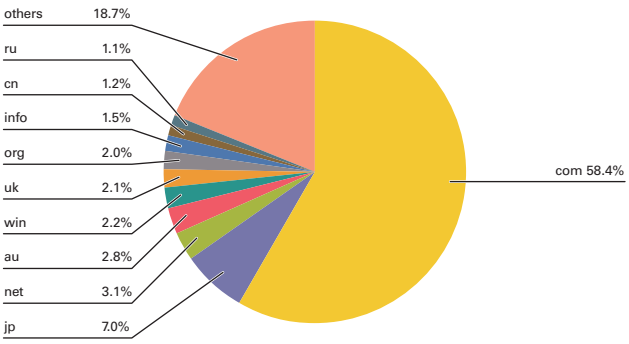


Figure 5: Ratio of Domain Names Authenticated Using DMARC by TLD

*5 Ministry of Internal Affairs and Communications: Statistical Data (http://www.soumu.go.jp/main_sosiki/joho_tsusin/d_syohi/m_mail.html#toukei) (in Japanese).
*6 DMARC, "Australian Government Agency Recommends DMARC, DKIM, and SPF" (https://dmarc.org/2016/08/australian-government-agency-recommends-dmarc-dkim-and-spf/).
*7 DMARC, "DMARC Required For UK Government Services By October 1st" (https://dmarc.org/2016/06/dmarc-required-for-uk-government-services-by-october-1st/).
*8 DHS, "Binding Operational Directive 18-01" (https://cyber.dhs.gov/bod/18-01/).

1.3.2 Implementation on the ".JP" Domain

Between April 2005 and May 2012, the WIDE Project entered into a joint research agreement with the Japan Registry Services (JPRS) that manages ".jp" domain names, and measured the deployment ratio of SPF and other technologies on these domains*9. This period coincided perfectly with the popularization of SPF, so the data proved invaluable for analogizing the degree of change and its effect.

Now the Ministry of Internal Affairs and Communications has decided to start these surveys again with DMARC added to promote its spread*10. As for specific methods, the Japan Data Communications Association, a subcontractor for the Ministry of Internal Affairs and Communications, will enter into a joint research agreement with JPRS. I am also taking part in these surveys as a visiting researcher for the Japan Data Communications Association.

According to the survey results published by the Ministry of Internal Affairs and Communications as of January 2018*5 (Table 1), the SPF configuration ratio of domain names used for email was 56.9% overall. Because the deployment ratio for SPF that we were investigating in the WIDE Project was the overall ratio of SPF records configured versus the number of domains where MX records were configured, the method of calculation is slightly different from the deployment ratio mentioned above. Using the same criteria, it would be 58.1% as of January 2018. For the WIDE Project, the deployment ratio was 43.89% as of May 2012, so using the same criteria, this represents an increase of about 14.2 percentage points.

This will be the first attempt to survey the configuration of DMARC records on ".jp" domain names. Although the adoption ratio proportional to incoming email (flow rate) on IIJ's email services was 19.3%, unfortunately, the actual ratio of configured ".jp" domains averaged out to just 0.6% of the total. The first SPF survey result produced by the WIDE Project was 0.1%, so we hope this ratio continues to grow. We have mentioned the initiatives by government agencies in the United Kingdom, Australia and the United States, and hopefully use will spread among government agencies in Japan as well. SPF actually has a high adoption ratio of 92.3% on the ".go.jp" domains used by government agencies in Japan. The ".lg.jp" domains often used by local government bodies also have an adoption ratio of 75.7%, which is the second highest by attribute type.

It is easier to configure DMARC records than SPF records, which require outbound email servers to be checked, so I believe if SPF records are already set up, we should start by configuring DMARC records with a "p=none" policy.

Attribute	No. of Domains	No. MX Set	No. SPF Set	SPF Set (%)	No. DMARC Set	DMARC Set (%)
AD.JP	252	212	140	66.0	6	2.8
AC.JP	3596	3367	2086	62.0	10	0.3
CO.JP	403955	380239	252961	66.5	1089	0.3
GO.JP	582	428	395	92.3	1	0.2
OR.JP	35146	33012	21043	63.7	71	0.2
NE.JP	13044	10617	5590	52.7	99	0.9
GR.JP	6112	5438	2884	53.0	27	0.5
ED.JP	5230	4852	2854	58.8	21	0.4
LD.JP	1652	1216	921	75.7	2	0.2
Geographic/Prefecture Type	13414	7530	3959	52.6	28	0.4
General-use	988365	756800	391728	51.8	5565	0.7
Total	1471349	1203711	684561	56.9	6919	0.6

Table 1: Survey Results for Sender Authentication Technology Configuration on ".JP" Domains

*9 WIDE Project, "Measurement Results on Deployment Ratio of Domain Authentications" (http://member.wide.ad.jp/wg/antispam/stats/index.html.en).

*10 Ministry of Internal Affairs and Communications, "Survey on the Configuration Status of Sender Authentication Technology on '.JP' Domain Names" (http://www.soumu.go.jp./menu_news/s-news/01kiban18_01000035.html) (in Japanese).

1.3.3 Trends in Standardization Including Related Technologies

So-called technical standards on the Internet such as DMARC are published in written form by the IETF (Internet Engineering Task Force)*11 as RFC (Request for Comments) documents. At the IETF, WGs (Working Groups) are created for each area under deliberation, and matters such as technical specifications are discussed within these WGs, ultimately leading to an RFC being issued. I attended the IETF 101 meeting held in March 2018, so here I report on recent IETF affairs and email-related circumstances.

IETF meetings are held three times a year. Generally, a venue in Europe, North America, or Asia is selected. The IETF 101 meeting was held in the European city of London. The next IETF 102 meeting is scheduled to be held in Montreal, Canada in July. It was reported that there were 1,189 attendees at the IETF 101 meeting. Conference rooms and time slots are organized in advance for each WG, and multiple WG meetings are held at the same time. Usually, each WG has a single meeting, but for WGs with many participants that are deemed to need more time for discussion, several meetings are held. Also, some WGs don't hold any meetings at all, so when you participate in an IETF meeting, you need to confirm the schedule ahead of time. Recently, more and more people have been participating online instead of attending in person. But if you have an opinion to share, it is best to be physically present at the venue.

At the IETF dmarc WG the DMARC specifications were issued as RFC 7489, and the specifications for ARC (Authenticated Received Chain) are currently being examined. Furthermore, improvements (mainly dealing with email redelivery issues etc.) to the DMARC Informational RFC that was issued are also being discussed to make it a Standards Track document, along with discussion about the information included in DMARC reports.

Other currently active WGs related to email include the dcrup WG that discusses cryptographic algorithms and additional key length for DKIM, and the jmap WG that is evaluating the new JMAP access protocol for replacing IMAP and SMTP using JSON format data.

In principle, anyone can participate in discussions at the IETF, which makes it possible to gather a wide range of opinions, but there do seem to be issues with technical specifications taking a long time to solidify. Discussions and reciprocal communication tests on email-related technology are carried out at M^{3}AAWG*12, so it seems that RFCs are formulated comparatively quickly in this case.

*11 IETF (https://www.ietf.org).
*12 M^{3}AAWG (https://www.m3aawg.org)

1.3.4 Legal Matters

To implement SPF, DKIM, and DMARC on the recipient side, it is necessary to reference email delivery information and email body text for authentication, so as a general rule you must get permission from email users to do this. Of these, SPF and DKIM enable the detection of large quantities of spoofed emails through sender authentication, so it was determined that under certain conditions the labeling of authentication results without prior consent is a legitimate business activity and can be deemed legal*13.

Meanwhile, with the new DMARC sender authentication technology, the domain administrator on the sending side can specify how to handle emails that fail authentication using the policy values set in the DMARC record. For example, this means if you set the policy to "p=reject", large quantities of spoofed emails are rejected and there is no need to deliver unwanted email to recipients. However, although legal matters regarding sender authentication technologies such as SPF and DKIM had been sorted out as far as the labeling of authentication results is concerned, this was not the case for processes such as DMARC where receipt is rejected.

In light of this,, we cleared up the remaining issues for implementing DMARC, including new methods for processing on the recipient side, through discussion centered around the Anti-Spam mail Promotion Council and other groups. The details have been published by the Ministry of Internal Affairs and Communications*13. These matters deal with recipient-side processing methods based on DMARC policies and DMARC reports. There are two types of DMARC report: aggregate reports and failure reports. For aggregate reports, comprehensive agreement means that individual consent is not required, enabling the recipient to transmit them to the sender. Failure reports, like error emails, include the details of the message that was originally sent. Consequently, it was determined that care should be taken when sending them to domain administrators that may not be party to the email communication. Therefore, when sending a failure report through comprehensive agreement, there is now a condition that it should not include the body or subject (content of the subject header) of the email that was originally sent. One of the aims of a failure report is to determine whether email that was sent actually failed authentication or is spoofed email. Even if the original outgoing email is not included in its complete form, it should be possible to determine whether it is legitimate email from other header information included in the failure report, so we believe the present limitations still provide sufficiently useful information.

We hope that clearing up these matters will help move the implementation of SPF, DKIM, and DMARC forward on the recipient side as well.

*13 Ministry of Internal Affairs and Communications, "Legal Matters Concerning the Sender Authentication, etc." (http://www.soumu.go.jp/main_sosiki/joho_tsusin/d_syohi/m_mail/legal.html) (in Japanese).

1.4 Conclusion

This year (2018) marks 10 years since the Anti-Spam mail Promotion Council, in which parties interested in spam countermeasures take part, was established*14. In 2014, LAP 10 Tokyo was held in Japan as the tenth annual meeting of LAP (London Action Plan, now UCEnet), an international government agency meeting on anti-spam measures. Also in 2014, the 10th anniversary meeting of M³AAWG, in which I have participated since its establishment, was held in Boston, United States.

In times gone by, 10 years seemed like a fairly lengthy period, although not an eon by any stretch. But like dog years, time seems to move faster in the IT industry these days, and 10 years now feels rather like a rather deep expanse of time. And yet the spam situation does not seem to have improved much at all, and as someone who has been involved in this industry for many years, I feel a fair amount of responsibility for this. At the same time, however, considering that in a worst-case situation email could become unusable, I still feel like I have made a contribution of sorts. Seeing how multiple chat applications are now being used with a focus on mobile devices, I also believe that the way communication tools are used will continue to change in the future. As I am in a position to consider new mechanisms such as these, I will continue to strive to address current email issues, evaluate new ways to communicate, and work to prevent similar issues from appearing there as well.

Author:
Shuji Sakuraba
Mr. Sakuraba is a Senior Manager of the Application Service Department of the Network Division, IIJ.
He is engaged in the research and development of communication systems. He is also involved in various activities in collaboration with external related organizations for securing a comfortable messaging environment.
He has been a member of M³AAWG since its establishment. He is acting chairperson of the Anti-Spam mail Promotion Council (ASPC) and a member of its administrative group, as well as chief examiner for the Technology Workgroup. Additionally, he is chairman of Internet Association Japan's Anti-Spam Measures Committee. He is also a member of the Email Security Conference program. Furthermore, he is a visiting researcher for the Japan Data Communications Association.

*14 Anti-Spam Consultation Center (https://www.dekyo.or.jp/soudan/contents/anti_spam/index.html) (in Japanese).

Issues with the Implementation of the RSA Algorithm Key Generation Module (ROCA)

2.1 Introduction

In October 2017, the scheduled publication of papers at ACM CCS 2017*1 triggered major news reports related to cryptographic technology. One was the report of an attack called KRACKs, caused by flaws in the WPA/WPA2 specifications*2. Because this was an issue with the protocol itself, fears that it would become a major problem sparked an overreaction on social networks. However, it could be fixed with a relatively minor correction, so these fears were unjustified. This incident was ranked fourth in the JNSA Major Security News*3, and it created quite a stir regarding the circulation of unverified information and appropriate methods of reporting.

Another was an issue called the Return of Coppersmith's Attack (ROCA), which was caused by the defective implementation of the key generation module in the RSA cryptographic algorithm*4. Although ROCA was reported at around the same time as the KRACKs attack, it did not receive much coverage in Japan. On the other hand, because it was an attack that enabled factorization of the RSA public key more practically in terms of attack time and cost, countermeasures were shipped promptly. It was recognized that this attack degrades functionality while hindering the expected performance of cryptographic functions, so vendors developed patches and prompted users to replace RSA key pairs created using the vulnerable key generation module. Other incidents*5 where bugs or design flaws have caused implementations of cryptographic modules to be far easier to compromise than they should be, or were thought to be, have been disclosed in the past, and ROCA would be listed as one of them. In this volume of the report, we examine past failures and give an overview of the factors behind vulnerabilities in cryptographic implementations that result from reduced space for keys and various parameters similar to ROCA. I will also touch upon the future impact of the results from a series of research projects on attacks such as ROCA.

2.2 An Overview of ROCA

Cryptographic technology is employed when using security protocols such as SSL or TLS. Public-key cryptosystems are used for encryption (ensuring confidentiality) and digital signatures (ensuring data integrity) in applications such as browsers, where general users can tell that a connection is secure when a locked key mark is displayed. One example is the RSA cryptosystem, which bases its security on the difficulty of prime number factorization. Most server certificates contain an RSA public key, and in SSL and TLS, for example, they are widely adopted as a mechanism for guaranteeing the validity of servers. Recently, from the perspective of perfect forward secrecy*6, encryption methods where an ephemeral key (temporary key) is generated using the DH or ECDH algorithm each time are recommended instead of using the public key stored in the server certificate to ensure confidentiality. This makes it possible to use a dedicated signature algorithm when a browser or user confirms whether a server is valid. A typical example of this is the ECDSA signatures*7 based on elliptic curve cryptography. In fact, a growing proportion of server certificates are issued containing ECDSA keys instead of RSA public keys, and these certificates are supported by major browsers. On the other hand, RSA-based server certificates are still currently in wide use, and would be affected by ROCA.

In October 2017, a research team at Masaryk University in the Czech Republic reported a vulnerability in RSA key generation modules made by Infineon Technologies AG and discussed its impact. The RSA algorithm is a cryptosystem where two primes generated at the time of key generation are used as a private key, and a composite number obtained by multiplying these two primes is used as a public key. Security is guaranteed by the fact that a vast amount of computation is required for factorization of the composite number that serves as the public key, meaning decryption is not practically possible. Now it has been reported that a vulnerability was discovered in an implementation of the RSA key generation module, and the bias in generated keys could

*1 ACM Conference on Computer and Communications Security 2017 (https://ccs2017.sigsac.org). CCS 2017 - Accepted Papers (https://acmccs.github.io/papers/).
*2 Key Reinstallation Attacks (https://www.krackattacks.com).
*3 JNSA, "Major Security News" (http://www.jnsa.org/active/news10/) (in Japanese).
*4 Centre for Research on Cryptography and Security (CRoCS), Masaryk University, "ROCA: Vulnerable RSA generation" (CVE-2017-15361) (https://crocs.fi.muni.cz/public/papers/rsa_ccs17).
*5 Internet Infrastructure Review Vol.17 "1.4.1 The Issue of Many Public Keys Used with SSL/TLS and SSH Sharing Private Keys with Other Sites" (https://www.iij.ad.jp/en/dev/iir/017.html).
*6 Internet Infrastructure Review Vol.22 "1.4.2 Forward Secrecy" (https://www.iij.ad.jp/en/dev/iir/pdf/iir_vol22_infra_EN.pdf).
*7 NIST, "FIPS 186-4 Digital Signature Standard (DSS)" (https://csrc.nist.gov/publications/detail/fips/186/4/final). The use of ECDSA is stipulated in Chapter 6. This document also covers DSA signature schemes in Chapter 4, and RSA signature schemes in Chapter 5.

enable factorization in a much shorter time than expected. There have been several reports of incidents where vulnerabilities were recognized due to keys or parameters being derived from a narrower space than originally intended, but most of these were caused by flaws in pseudo-random number generation modules. Although the current issue could be categorized as similar if you only look at the research results, the true nature of the problem actually lies not in the pseudo-random number generator but in a flaw in the implementation of the part related to prime number generation.

2.3 Key Lifecycles and Previous Failures

Two types of keys are used in public key cryptosystems: a private key used for creating digital signatures and decryption, and a public key made available through a server certificate or key repository. Thus, users first generate key pairs prescribed by each cryptosystem, and when doing this process the key pairs are generated using a random number sequence that is derived from a pseudo-random number generation module and can be safely used as a source. This pseudo-random number generation module is deployed so that it can be used as necessary to obtain random number sequences both at the time of key generation and when keys are used (for example, at the time of signing). When using keys—such as when performing encryption with a public key, signing with a private key, or verifying a signature—an expiration date is often set for the public key. After a key expires, it is discarded and a new one is generated, so it has a lifecycle. This key management flow also includes a path for forcibly invalidating the key even before it expires if the corresponding private key leaks or may have leaked. SSL/TLS server certificates have an expiration date, and if you imagine a system where certificates are discarded before this date, you should be able to grasp this concept of lifecycles.

Next, we will look at a few incidents of failure to understand at which stage of the abovementioned key management lifecycle the problem occurred. One case similar to this one where problems occurred when generating RSA keys is a key generation issue in Debian OpenSSL that was disclosed in 2008*8. When using OpenSSL to generate keys in a specific version of Debian, there was a bug where private keys were derived from an extremely small key space. In this incident, a list of all public keys that could be generated from the vulnerable key generation module was released so that anyone could check their keys.

Another similar key generation problem was a flaw in Taiwan citizen cards*9 that was identified in 2013. These were IC cards that had passed the accreditation criteria of a cryptographic module called FIPS 140-2, but significant bias was seen in the prime number generator, and there have been reports of public keys that enabled efficient factorization. Unlike ROCA, this is categorized as a flaw in the pseudo-random number generation module. The vulnerable keys reported here include multiple examples of an issue reported in 2012 where private keys were shared unintentionally*10, enabling factorization with far less computational complexity. Cases where private keys were unintentionally shared were due to the fact that the key space was small, with an implementation for a certain IC card, for example, only able to generate a mere 36 different RSA public keys, and these reports had an extremely large impact*11*12.

There was also a similar case of failure at the time of keys being used rather than at the time of key generation. In this case, an Android application incorporating a Bitcoin wallet function reused the parameters employed for ECDSA signatures, enabling the private key to be identified from two different signatures belonging to the same entity*13. This implementation ignored the restriction that parameters should not be reused as a signature algorithm. Specifically, this was caused by the corresponding parameters overlapping by chance because the pseudo-random number generation module used by the Android application had low entropy outputs. As this demonstrates, issues similar to ROCA are occurring in various phases.

*8 VU#925211, “Debian and Ubuntu OpenSSL packages contain a predictable random number generator” (https://www.kb.cert.org/vuls/id/925211).
*9 Daniel J. Bernstein et.al, “Factoring RSA keys from certified smart cards: Coppersmith in the wild,” Cryptology ePrint Archive: Report 2013/599 (https://eprint.iacr.org/2013/599).
*10 PKI Day 2012, Y.Suga, “The Issue of Many Public Keys Unintentionally Sharing Private Keys with Other Sites” (http://www.jnsa.org/seminar/pki-day/2012/data/PM02_suga.pdf) (in Japanese).
*11 Arjen K. Lenstra et al., “Ron was wrong, Whit is right” (https://eprint.iacr.org/2012/064).
*12 Nadia Heninger et al., “Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices,” Proceedings of the 21st USENIX Security Symposium (https://factorable.net/paper.html).
*13 Bitcoin Project, “Android Security Vulnerability” (https://bitcoin.org/en/alert/2013-08-11-android).

2.4 The True Nature of the ROCA Issue

It could be said this issue that was found in a cryptographic library made by Infineon Technologies AG has the same root cause as other examples, such as the Debian OpenSSL bug, in that the generated key space is too small. However, it should be noted that this is due to problems with the key generation algorithm, rather than bias in the random data output from the pseudo-random number generation module.

Researchers claim that the vulnerability in the key generation module was not discovered through a source code review or reverse engineering. It came to light that the key space that can be established is small when bias is observed in key data after treating the key generation module as a black box and generating a ton of key pairs. RSA public keys are obtained from the product of two primes. The processing of a typical key generation module has a step for determining whether the odd numbers generated from random data as candidates are primes. Because these primality tests generally take time, they can become a bottleneck when generating keys in environments such as IC cards that have limited computation speed and memory space. We know that prime numbers generated by the vulnerable key generation modules discovered in this case are distinctive in that they are derived from a much smaller space than the full prime number space. As identified in the paper by the researchers, it is thought that the constraints of this distinctive prime number format were intended to accelerate prime generation. It seems the people who designed and implemented it did not realize that using this sort of acceleration would create a vulnerability. It is conceivable that a high bar was set for response time, and processing that should be carried out was omitted because performance fell far short of the processing goal.

Similar issues include the fact that each year there have been several dozen vulnerability reports indicating that certificate validation is omitted when conducting SSL/TLS communications in the background in Android applications. Typical browsers notify the user of the certificate validation results via the URL input field or security indicator when communicating with SSL/TLS servers. However, because this sort of information display or user action isn't always necessary for SSL/TLS communications carried out in the background on an Android app, we believe the decision was made to omit the certificate validation module to speed up processing.

The prime number p generated by the vulnerable module identified in ROCA has been found to have the following characteristics:

$$p = k*M + (65537^{a} \bmod M)$$

Here M is the product of n consecutive primes from 2, determined by the length of the prime you want to generate (i.e. n = 39 for 256 bits, and n = 71 for 512 bits). Because M is automatically fixed when n is fixed, prime number candidates are selected by moving parameters k and a as appropriate. For example, the generation of an RSA-512 public key is achieved by multiplying two 256-bit primes. According to the prime number theorem, which indicates the density of primes, it is known there are about $2^{248.5}$ candidates, so we can see it is possible to randomly select a prime from an extremely large prime number space. Meanwhile, in this vulnerable prime generation module with n = 39, M = 2*3*5*...*167 is about 219 bits long, so only 37 bits can be moved via parameter k. Similarly, only 62 bits can be selected via parameter a, resulting in just 99 bits worth of entropy. This means that primes are generated from a considerably smaller key space than usual.

RSA is an algorithm that bases its security on the difficulty of factorization, and the NIST has estimated the equivalent encryption strength in symmetric-key cryptography key bit length when using various RSA key lengths. For example, their tables show that 2048-bit RSA has 112 bits of security strength, and elliptic curve cryptography such as the aforementioned ECDSA has a security strength of exactly half its key length*14. With incidents such as the factorization of 768-bit RSA keys in 2010, the use of RSA keys of at least 2048-bit length is currently recommended. When using RSA signatures with PKI, organizations such as the CA/Browser Forum now prescribe a policy of migrating away from the use of 1024-bit RSA keys, as well as the SHA-1

*14 NIST, SP 800-57 Part 1 Rev. 4, "Recommendation for Key Management, Part 1: General" (https://csrc.nist.gov/publications/detail/sp/800-57-part-1/rev-4/final).

hash function*15 recognized as one of the compromised algorithms after collisions were discovered last year. Such policy is now followed when certificates are issued. A report published each year at CRYPTREC gives comparisons with the processing capability of supercomputers*16, and this supports the fact that 2048-bit RSA is sufficiently secure at this point in time.

2.5 The Impact of ROCA

As mentioned in the previous section, it is easy to see how the search space for factorization becomes smaller because the key space narrows significantly due to constraints on generated primes. Similarly, there is research that aims to make factorization easier through the use of the form of private key constraints, such as the well-known Coppersmith method*17 that appears in the title of ROCA. The Coppersmith method can efficiently restore p from product N of the two primes p and q (i.e. N = p*q) used in the RSA algorithm and the blobs of the lower half of the prime p enable successful factorization as a result. It is also one of the efficient methods that derived the private key of SSL/TLS servers in a competition designed to determine what level of threat the Heartbleed bug in OpenSSL posed*18.

By extending the Coppersmith method to ROCA, the cost of using the amount of cloud resources required for factorization is estimated to be as shown in Table 1. We can see that even the 2048-bit RSA keys in wide use today can be decrypted with practical low costs. Within just a week of the announcement of these results, it was suggested that factorization could be achieved with 5–25% more efficient costs. This means that factorization is even easier and less costly than estimated in the original paper*19.

Tools for checking whether RSA keys were generated by the affected cryptographic modules were released by the authors. Various verification methods were made available, including Python code*20 that lets you perform checks offline, an online version that lets you check by posting public keys via a browser*21, and a version that lets you send an S/MIME signature via email to obtain results. The system for checking for the ROCA vulnerability is very simple and concise*22. According to the authors there are no false positives, and the probability of accidentally generating keys vulnerable to ROCA is just 2^{-154}, which is negligible.

It has been announced that the certificates used with e-Residency ID cards*23 issued by the government of Estonia are affected by ROCA*24, and users are being asked to perform firmware updates and re-generate keys*25*26. According to the original paper, around 4,400 e-Residency ID cards were surveyed, and it was shown that all of them were affected. Also, the ROCA bug has been present since 2012, and although they would have expired by now, it is thought there are keys that continued to be used

Table 1: The Costs When Using the Cloud Resources Required for Factorization

RSA key length	CPU resources required for factorization	Factorization costs when using the cloud
512 bit	1.93 CPU hours	$0.06
1024 bit	97.1 CPU days	$40–$80
2048 bit	140.8 CPU years	$20,000–$40,000

*15 CRYPTREC Cryptographic Technology Guideline - SHA-1 (https://www.cryptrec.go.jp/topics/cryptrec_20180427_eval_gl_2001_2013r1.html) (in Japanese). Security Diary, SHAttered attack (SHA-1 collision discovery) (https://sect.iij.ad.jp/d/2017/02/271993.html) (in Japanese).
*16 CRYPTREC Report (http://www.cryptrec.go.jp/english/report.html).
*17 Don Coppersmith, "Finding a Small Root of a Bivariate Integer Equation; Factoring with High Bits Known", EUROCRYPT96, 178–189.
*18 IIJ-SECT Security Diary, "Regarding the feasibility of private keys leaking due to the Heartbleed bug" (https://sect.iij.ad.jp/d/2014/04/159520.html) (in Japanese).
*19 The cr.yp.to blog, "2017.11.05: Reconstructing ROCA" (https://blog.cr.yp.to/20171105-infineon.html).
*20 ROCA: Infineon RSA key vulnerability (https://github.com/crocs-muni/roca/).
*21 ROCA Vulnerability Test Suite (https://keychest.net/roca), Test your RSA Keys (https://keytester.cryptosense.com).
*22 Key fingerprinting (https://github.com/crocs-muni/roca/blob/master/roca/detect.py).
*23 Republic of Estonia, "e-Residency" (https://e-resident.gov.ee/become-an-e-resident/).
*24 Politsei.ja Piirivalveamet, "Possible Security Vulnerability Detected in the Estonian ID-card Chip" (https://www2.politsei.ee/en/uudised/uudis.dot?id=785151).
*25 Politsei.ja Piirivalveamet, "For the user of ID-card and mobile ID" (https://www2.politsei.ee/en/nouanded/isikut-toendavad-dokumendid/id-kaardi-ja-mobiil-id-kasutajale.dot).
*26 Politsei.ja Piirivalveamet, "Renewal of document certificates-frequently asked questions" (https://www2.politsei.ee/en/teenused/isikuttoendavad-dokumendid/sertifikaatide-uuendamine/).

for over five years without the vulnerability coming to light, despite the fact that factorization was possible from the beginning. Each company, including Japanese vendors, has provided detailed information on the products enabling TPM (Trusted Platform Module) chips*27*28. The recommended actions were to update the firmware and discard any RSA key pairs generated by the vulnerable TPM chips, then create new ones. TPM is a tamper-resistant chip that protects against physical attacks on private keys, and it had been regarded as more secure than storing key data in memory or other storage devices. However, with the discovery of ROCA, new potential disadvantages of making a module that uses keys as a black box were revealed. While outsourcing parts that are difficult to manage lets you take a more hands-off approach, new threats may arise in areas no longer under your control.

2.6 How ROCA Was Discovered

As we saw in section 2.4, the ROCA vulnerability was discovered by focusing on the prime generation modules that output special form keys. The process that discovered a bias by observing a large volume of primes without reverse engineering or accessing the source code is based on prior research by the same team. This was a study of the RSA keys generated by 38 cryptographic software products and IC cards presented at USENIX Security 2016, which was held in August 2016*29. The study revealed that each cryptographic library had distinctive features using heat maps that plotted primes p and q on the two axes to show the frequency distribution of seven bits from 2nd to 8th most significant. It also organized the 38 products into 13 categories by investigating trends through extraction of just the most distinctive nine bits, covering a prime's 2nd to 7th most significant bits, the 2nd least significant bit, prime mod 3 (as a value), and public key N mod 2 (as a value). The Infineon module targeted by ROCA was categorized as Class 12, and at this point it was identified that it had more distinctive characteristics than the cryptographic libraries belonging to other categories.

Related research results were also released by the same research team at ACSAC 2017, which was held after ACM CCS*30. This was an attempt to identify the cryptographic library that generated a public key from just the public key information using their prior knowledge of cryptographic libraries. The same approach was used at USENIX Security 2016 to identify the cryptographic library by calculating the remainder of mod 3 and mod 4 for public key N and detecting whether there was bias. The researchers claim there are one percent or fewer false positives. They also pointed out the impact on privacy issues. For example, observing the public key information used in Tor reveals that different nodes are identified as the same user or specified region.

2.7 The Potential Impact of ROCA on Other Cryptographic Algorithms

Because ROCA itself is a vulnerability in the key generation module for the RSA cryptosystem, or in other words an issue with prime generation logic, it is unlikely to affect cryptographic algorithms other than RSA that generate and use primes but do not need to keep them secret. In the RSA cryptosystem, for example, two 1024-bit primes are required as private key candidates when generating a 2048-bit RSA public key. Meanwhile, the private keys for the abovementioned ECDSA signatures used in Bitcoin can be generated simply by deriving an integer (256-bit length), so no complex logic is required. In other words, only the validity of the pseudo-random number generation module affects the security of the key generation module, so it is unlikely that a vulnerability like ROCA will come into play.

In the cryptographic key lifecycle, we consider the phase in which the key will be used and not key generation. It is normally necessary to generate and sign different parameters each time, as seen with the aforementioned problems of pseudo-random number generation modules in Android, but the special logic seen in the prime generation process is not required here either. However, in the generating process of private keys or various parameters, considering the cases where random data is derived by

*27 Infineon Technologies AG, "Information on TPM firmware update for Microsoft Windows systems as announced on Microsoft's patchday on October 10th 2017" (https://www.infineon.com/cms/en/product/promopages/tpm-update/?redirId=59160).

*28 CERT, "Vulnerability Note VU#307015, Infineon RSA library does not properly generate RSA key pairs" (https://www.kb.cert.org/vuls/id/307015).

*29 Centre for Research on Cryptography and Security (CRoCS), Masaryk University, "The Million-Key Question - Investigating the Origins of RSA Public Keys" (https://crocs.fi.muni.cz/public/papers/usenix2016).

*30 Centre for Research on Cryptography and Security (CRoCS), Masaryk University, "Measuring Popularity of Cryptographic Libraries in Internet-Wide Scans [ACSAC 2017]" (https://crocs.fi.muni.cz/public/papers/acsac2017).

simply filling the most significant bits with zeros or repetition of a short bit string, we cannot ignore the potential for vulnerabilities caused by a small data space even in the ECDSA signature algorithm. At this time it is possible to check whether a public key in the Bitcoin network has derived the same key pair by referring to the blockchain history. In other words, it is possible to determine whether you have shared private keys with other third parties. As such, it is conceivable that there exist implementations in which a backdoor that makes the key space smaller has been incorporated even though it appears on the surface that keys are generated according to proper procedures.

If it is unintentionally revealed that a key generation module is vulnerable, as with ROCA, this could lead to attacks that share key pairs with another party with a higher probability than expected when repeatedly generating keys. This has no effect on cold wallets that ensure security by not connecting signing keys themselves and/or signature modules to the network. This demonstrates how crucial the key generation phase is for all virtual currencies, not just Bitcoin. As seen in the abovementioned example of the Taiwan citizen cards, even cryptographic libraries or HSMs (Hardware Security Modules) certified FIPS 140-2 require key management that takes into account the fact that vulnerable products may exist. In the Bitcoin network, Bitcoin addresses are used as owner IDs. Bitcoin addresses are created by a generated key pair using the ECDSA signature method on the elliptic curve known as secp256k1. Then, after calculating the digest from the public key data with version information and checksum data, using two different hash functions, this is finally converted into a human-readable string of 26 to 35 characters using Base58 encoding. There are known methods for deriving a "desired Bitcoin address" so that characters specified by the user appear in the Bitcoin address during this procedure. The hash functions used here are SHA-2 and RIPEMD-160, and because it is difficult to derive the intended output from them, different key pairs are derived through trial and error repeatedly. This method generates a significant number of key pairs, so secure random data is required here as well. Many generation tools such as these are now available, but there is no way to trust them, and as they are distributed as binary rather than source code in some cases, they must be used with caution.

Here we discussed vulnerabilities in cryptographic modules in which the RSA algorithm is implemented, along with privacy issues that could occur in the future. It was reacknowledged that implementations of cryptographic algorithms are susceptible to bugs because its prime number generation and primality tests require slightly abstruse logic, but the RSA algorithm itself is mostly unscathed, and it can be used securely as long as knowledge regarding its implementation is shared. With the expectation that it will no longer be usable in the future when factorization becomes trivial due to the advent of quantum computers, next-generation algorithms known as post-quantum cryptography are also being developed*31. NIST is currently conducting a standardization competition, and the anticipated schedule involves cryptanalysis and evaluation over the next three to five years, followed by the sharing of a standardized draft within another two years or so*32. When implementing next-generation cryptographic technology, there may be mechanisms that designers and implementors find more complicated and situations that require a vast quantity of knowledge to understand. Comprehending the true nature of issues like ROCA that occur on modern cryptosystems such as RSA should serve as a lesson for the next generation.

Author:
Yuji Suga
Senior Engineer, Office of Emergency Response and Clearinghouse for Security Information, Advanced Security Division, IIJ
Dr. Suga has been in his current position since July 2008. He is engaged in investigation and research activities related to cryptography and information security as a whole. He is a member of the CRYPTREC Cryptographic Technology Promotion Committee.
Dr. Suga also serves as secretariat of the Cryptographic protocol Evaluation toward Long-Lived Outstanding Security Consortium (CELLOS). He is assistant secretary of the ISEC Technical Committee of the Institute of Electronics, Information and Communication Engineers (IEICE). He is an organizing committee member for IWSEC2018.
He is an organizing committee member for ECC2018. Virtual Currency Governance Task Force (VCGTF) Security WG member.

*31 Internet Infrastructure Review Vol.31 "1.4.3 Trends in Post-Quantum Cryptography" (https://www.iij.ad.jp/en/dev/iir/pdf/iir_vol31_EN.pdf).
*32 Dustin Moody, "THE SHIP HAS SAILED: The NIST Post-Quantum Crypto 'Competition'" (https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/asiacrypt-2017-moody-pqc.pdf).

Countermeasures against Transmission of Illegitimate Emails on Large-Scale Email Systems

3.1 Introduction

For many years, IIJ has offered large-scale email system construction and email ASP services for service providers that have anywhere from hundreds of thousands to millions of users. Operating a large-scale email system requires considerable know-how, and in particular significant knowledge and experience is needed to incorporate effective countermeasures against the improper use of email. Since most of the work performed by email system administrators involves responding to improper use of the email system and associated troubleshooting, systematically implementing thorough countermeasures can reduce the workload while also contributing significantly to providing stable services to end users.

Trends in improper use shift on a daily basis, so email system administrators end up locked in a game of cat-and-mouse as they deal with each issue as it arises. Even if you consistently deal with improper use, it is difficult to reduce it, and this tends to be a futile struggle that is physically draining. On the other hand, it is possible to curtail improper use considerably by implementing mechanisms that allow you to focus systematically on key points.

Here we examine and explain several countermeasures against the improper use of email that have been effective, based on experience implementing and operating them on a variety of email systems. The countermeasures described here include those that require service providers to define a services agreement and obtain user consent, so we must note that individual service providers will need to consider whether each of the measures is appropriate for them.

3.2 The Importance of Countermeasures Against the Improper Use of Email

When an end user sends email using a service provider's email server, generally either SMTP authentication is performed by the MUA or Webmail is used. Although an authentication ID and password are necessary for SMTP authentication and the use of Webmail, if the user sets a password that is easy to guess, or if the PC itself is infected with a virus, these credentials are more likely to be leaked. There are many cases where spam is sent by impersonating a legitimate end user using authentication IDs and passwords leaked in this manner.

Spam generally involves the repeated automatic transmission of large volumes of email. As a result, email systems are flooded with email, which is ultimately sent over the Internet. When a large volume of spam is sent over the Internet, the outgoing IP address of the email system is recognized as the spam source and added to a blacklist on the Internet side. Being added to a blacklist has the following kinds of effects (Figure 1).

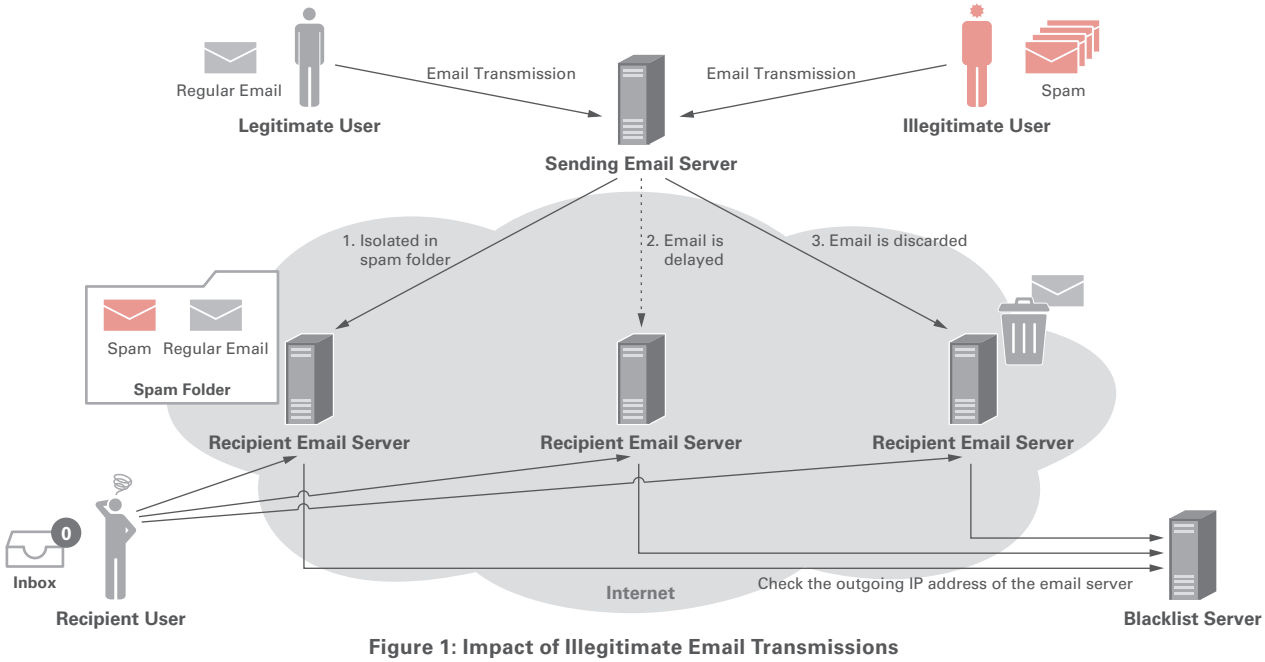


Figure 1: Impact of Illegitimate Email Transmissions

1. Regular email is detected as spam (in many cases Gmail, Hotmail, Yahoo.com, or a mobile phone carrier email service).
2. Email is delayed.
3. Email is discarded, and not delivered at all.

To be removed from a blacklist, it is necessary to eliminate the cause and restore normal service. However, many aspects of investigating the cause and implementing countermeasures involve trial and error, and this process can be quite exhausting for system administrators, so effective countermeasures against the improper use of email are a crucial part of system operations.

3.3 Trends in Illegitimate Email Transmissions

The nature of illegitimate email transmissions changes from moment to moment. Based on current trends, most illegitimate email is sent from overseas locations, as in the following examples.

1. A large volume of email is sent from overseas using a single authentication ID.
2. A large amount of email is sent simultaneously from overseas using multiple authentication IDs.
3. A large volume of email is sent from overseas using Webmail.

Basically, because all of these involve email being sent from overseas locations, it is possible to implement effective countermeasures against the sending of illegitimate email if a mechanism for determining the country of origin based on the email source IP address is available.

To implement such a mechanism, it is necessary to create a database that distinguishes countries based on the source IP address, along with a system that enables this to be used easily. There are several varieties of country database, such as MaxMind GeoIP2, but each differs in terms of whether they cost money or are free, whether support is available, whether they contain information other than country designations (regional level designations, Gmail's range discrimination, etc.), and whether they are updated frequently, so it is necessary to select one that matches your needs.

There are also ways to use country databases via API or by downloading and formatting them. Large-scale email systems on a service provider level tend to need to refer to country databases more often, because they receive a large volume of email, and

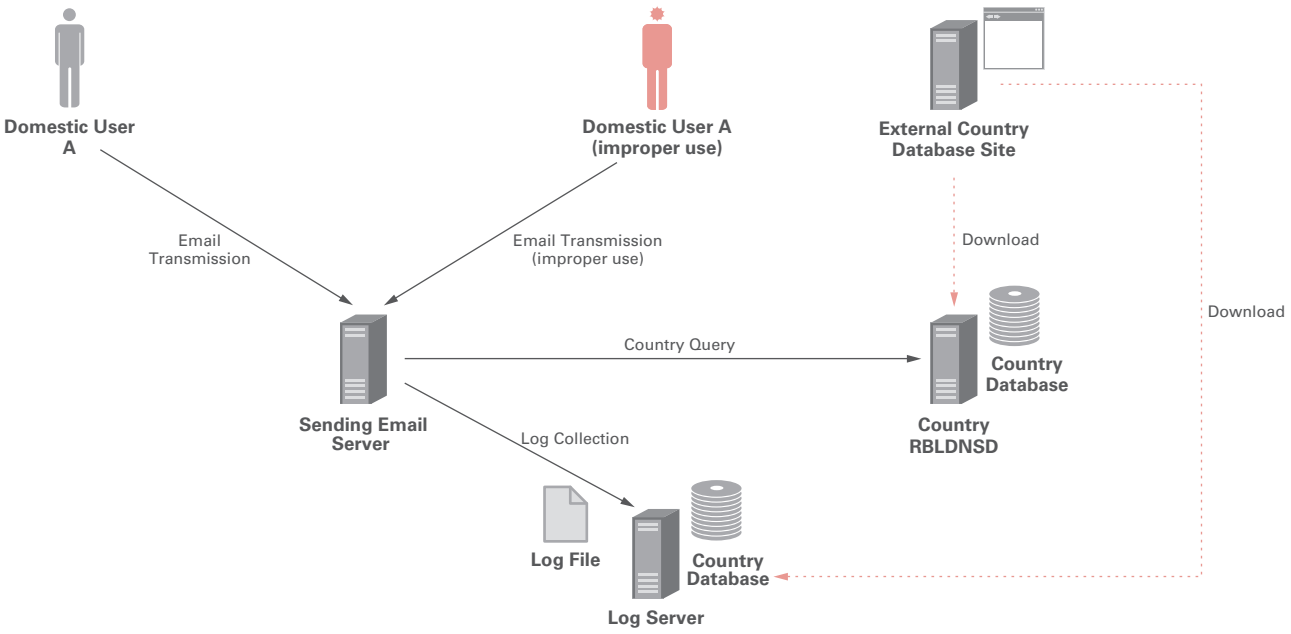


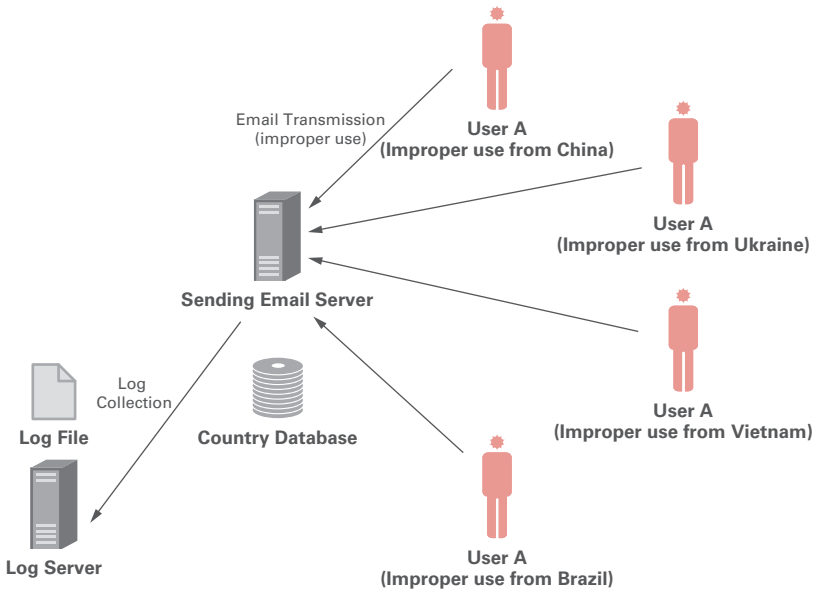
Figure 2: Overview of Country Data Detection System

based on experience the country data do not need to be updated very often, so we believe that downloading and shaping the data is a suitable approach in this case. The downloaded country databases are used for log analysis and real-time identification of countries on the email server, which I will explain later. Figure 2 shows a sample configuration.

3.4 Detecting Overseas Sources Using Email Logs

Typically, email server logs contain the authentication IDs and source IP addresses used for SMTP authentication. Creating a program to analyze email server logs using the aforementioned country databases enables analysis of the number of transmissions by country and the total number of emails sent for each authentication ID, making it possible to easily identify illegitimate email (Figure 3).

- **When email is sent from multiple countries within a few hours**
  For example, if emails are sent from China, Ukraine, Vietnam, and Brazil at approximately the same time using the same authentication ID, because there is no way for the user to be in different places around the world simultaneously, this makes it easier to determine that the authentication ID has been used illegitimately and take action to stop email being sent from these sources.
- **When a large volume of email is sent from a single overseas country**
  There are still many cases where a large volume of email is sent from a single overseas country. In this case, unlike the previous one, it may be hard to determine whether it is clearly an illegitimate use of email. However, if the number of emails sent within a certain period greatly exceeds a commonsense amount, it may be necessary to temporarily suspend email transmissions from the source in question. There are also cases where a large volume of legitimate email is sent from an overseas office, so adding certain authorization IDs to a whitelist is also something that must be considered.



● Log for email transmission from China (example of connection from China IP address: 1.0.*.*** using the authorization ID: example@example.com)
May 25 03:34:09 server11 smmta[16316]: AUTH=server, relay=from.example.com [1.0.*.***] (may be forged), authid=example@example.com, mech=PLAIN, bits=0
May 25 03:34:09 server11 smmta[16316]: w4OIY9On016316: from=from@example.com, size=0, class=0, nrcpts=1, proto=ESMTP, daemon=MSA, tls_verify=NONE, auth=PLAIN, relay=from.example.com [1.0.*.***] (may be forged)

● Log for email transmission from Brazil (example of connection from Brazil IP address: 23.97.*.*** using the authorization ID: example@example.com)
May 25 03:35:23 server11 smmta[16319]: AUTH=server, relay=from.example.com [23.97.**.***] (may be forged), authid=example@example.com, mech=PLAIN, bits=0
May 25 03:35:23 server11 smmta[16319]: w4P5Y60n028961: from=from@example.com, size=0, class=0, nrcpts=1, proto=ESMTP, daemon=MSA, tls_verify=NONE, auth=PLAIN, relay=from.example.com [23.97.**.***] (may be forged)

Figure 3: Detecting Improper Use Through Log Analysis

This sort of log-based approach is a versatile and effective method because it can be applied to the majority of email systems. However, the batch processing method often used for logs hinders the ability to respond in real time, and in some cases, systems have been hit with 10,000 or more emails before illegitimate transmissions were suspended, resulting in senders being added to a blacklist. This means that for large-scale email systems, there are times when simply analyzing email logs is not effective enough.

3.5 Implementing Real-Time Overseas Country Detection on Email Servers

There is a way to detect the originating country in real time by having the email server query a country database (such as an independently built RBLDNSD for countries) with the source IP address of the sender (Figure 4). Because this method enables real-time detection at the email server level, it is possible to identify email sent from multiple countries within the same time period as illegitimate and immediately suspend transmission. This enables you to eliminate the delay before a suspension is imposed, which was the issue with email log analysis. As a result, you can prevent situations where you only notice large volumes of email after it is sent, so implementing this solution can have a significant impact.

By fine-tuning the implementation, it is also possible to apply suspensions to only email from overseas and not the entire system, reducing the impact on end users whose usage takes place predominantly in Japan. This also offers a superior experience from the perspective of user support.

That said, implementing a real-time detection function on an email server requires the introduction of a Milter program, the modification of OSS email servers such as Postfix and Sendmail, or the use of a commercial email server with programming functions (such as Cloudmark Security Platform for Email or Vade Secure). Although the benefits of implementation are significant, the high degree of technical difficulty involved is a concern to some. In fact, when implementing this solution in the past, carefully performing repeated tests to ensure quality required a considerable amount of time.

3.6 Countermeasures Against Transmission of Illegitimate Emails via Webmail

The abovementioned countermeasures against the transmission of illegitimate emails were aimed at the sending of email using SMTP authentication, but the transmission of illegitimate email by Webmail is also on the rise. Several typical varieties of Webmail software are available in the Japanese email industry, and although each uses completely different methods to perform login and email transmission, the fact is that large volumes of email have been sent in formats corresponding to each software application. It has also been reported that the outbox is emptied and the user logged out once the emails have been sent so that no trace of the email having been sent via Webmail remains, and my impression is that this sort of improper use is becoming significantly more sophisticated.

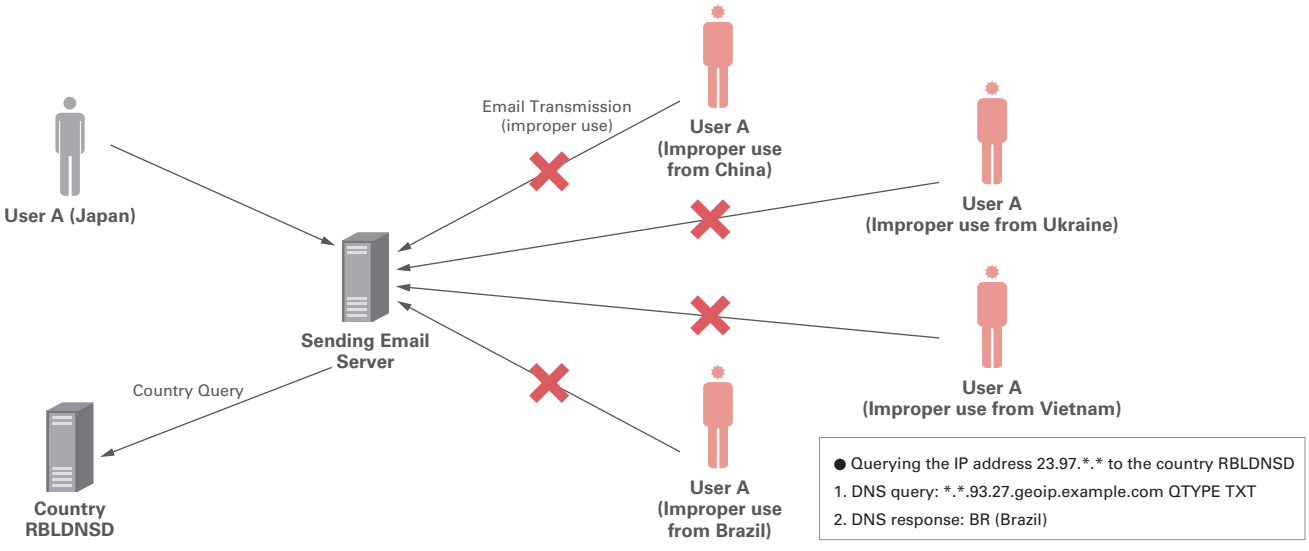


Figure 4: Real-Time Detection of Improper Use

Generally, because the recipient email server does not know the source IP address of email transmitted via Webmail, the country detection detailed above is hard to perform. As a result, it is difficult to detect improper use, and therefore Webmail is targeted by attackers as a secondary method of sending illegitimate email. To counter this, we adopted Webmail software that can embed source IP addresses in the email headers, and we also implemented controls using header information on the email server side (Figure 5). By taking this approach, the email server is able to detect countries in real time, making early detection and countermeasures against improper use possible. Webmail is also modified to prevent accounts affected by improper use from using it, or to suspend its use from overseas.

3.7 User Control over the Use of Email from Overseas Locations

You can also leave control of email access from overseas locations up to the user (Figure 6). Specifically, this method involves having users select whether to allow the transmission of email using SMTP authentication, the transmission of Webmail, and the receipt of POP/IMAP email from overseas, applying access control on a user-by-user basis on the email server side.

This makes it possible to restrict the transmission and receipt of email from overseas as well as Webmail logins under normal circumstances, while also providing the ability to enable these options from the management screen when on an overseas business trip or vacation.

This method is useful because giving end users the choice makes it easier to obtain their permission, while also serving to raise awareness of how access from overseas can easily lead to improper use. That said, as it is necessary to get consent from users to restrict access from overseas by default, it may not significantly reduce the improper use of email.

3.8 Countermeasures against SMTP Connection DoS Attacks

To stray from the topic a bit, one example of illegitimate email transmissions is where email server connections are deliberately exhausted by directing a high volume of SMTP connections at the server for a sustained period using multiple authentication IDs. This is a sophisticated type of DoS attack that identifies and exploits the timeout mechanism of email servers. These attacks

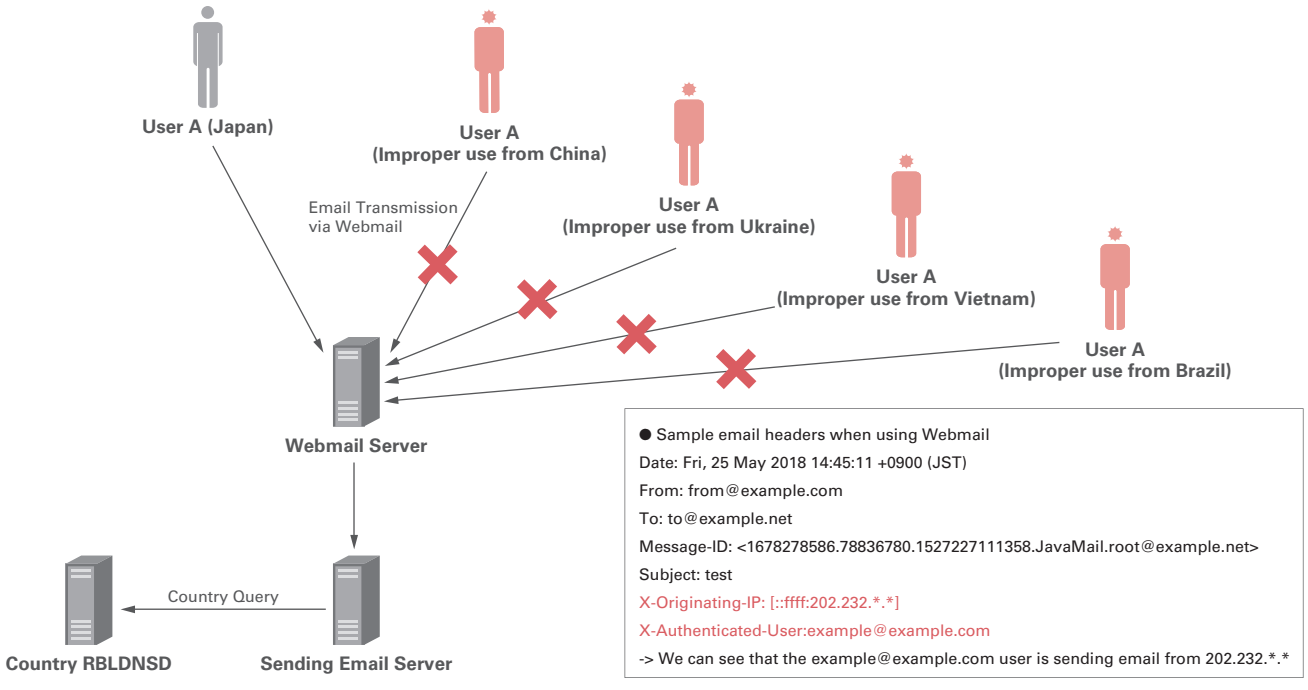


Figure 5: Detecting Illegitimate Email Transmissions via Webmail

frequently prevent new SMTP connections from being made at all, which has a significantly adverse impact on the provision of services (Figure 7).

One common method for dealing with sustained SMTP connections is to reduce the timeout period for the email server. Because the standard timeout period for email servers is often long, when deploying a server it is necessary to fine-tune the timeout period appropriately. Meanwhile, RFC 5321 (Simple Mail Transfer Protocol) defines recommended values for SMTP timeout under 4.5.3.2 Timeouts. Timeouts cannot be shortened in many cases because careless configuration can impact the transmission of legitimate email, so there are limitations to adjusting them.

Other methods include periodically checking the number of SMTP connections, and when this number approaches the system's limit, determining which sessions have a long idle timeout and disconnecting them from the server side (using the tcpkill command, etc.). By disconnecting SMTP sessions targeted at accounts affected by improper use, this also makes it more difficult for SMTP sessions to accumulate, reducing the risk of SMTP connection DoS attacks.

3.9 Conclusion

There are many measures for counteracting the transmission of illegitimate emails on large-scale email systems, so you need to combine multiple methods effectively. Implementation is costly and requires technical expertise, but considering the significant reductions to email service maintenance and system administrator workload, along with increased motivation, we believe that some form of countermeasures against the transmission of illegitimate emails are essential. I hope this article serves to aid the stable operation of email systems.

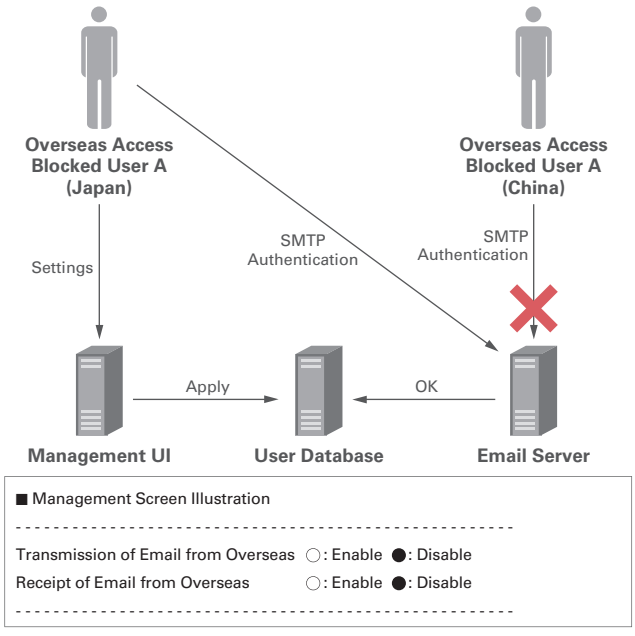


Figure 6: User Control over the Use of Email from Overseas Locations

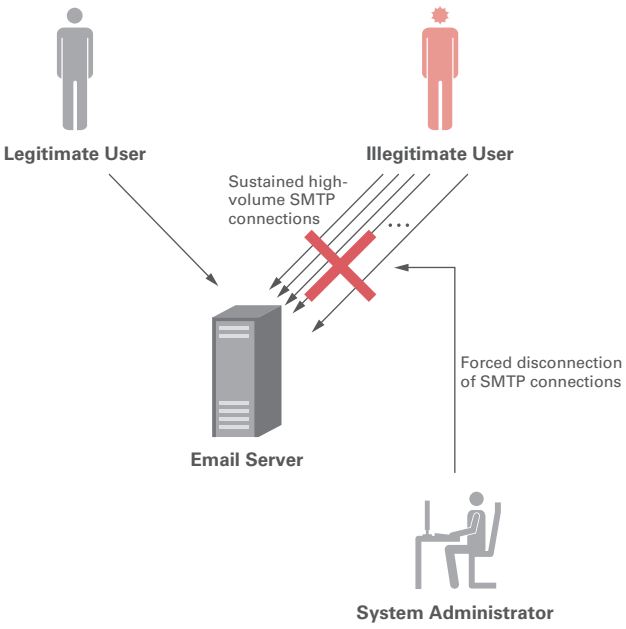


Figure 7: SMTP Connection DoS Attacks

Author:
Shigehiro Kinugasa
Manager, Mail Solutions Section, Enterprise Solutions Department, Cloud Division, IIJ

June 2018 Vol.39

IIJ
Internet Initiative Japan

About Internet Initiative Japan Inc. (IIJ)

IIJ was established in 1992, mainly by a group of engineers who had been involved in research and development activities related to the Internet, under the concept of promoting the widespread use of the Internet in Japan.

IIJ currently operates one of the largest Internet backbones in Japan, manages Internet infrastructures, and provides comprehensive high-quality system environments (including Internet access, systems integration, and outsourcing services, etc.) to high-end business users including the government and other public offices and financial institutions.

In addition, IIJ actively shares knowledge accumulated through service development and Internet backbone operation, and is making efforts to expand the Internet used as a social infrastructure.

IIJ-MKTG020-0037

Internet Initiative Japan Inc.

Address: Iidabashi Grand Bloom, 2-10-2 Fujimi, Chiyoda-ku,
Tokyo 102-0071, Japan
Email: info@iij.ad.jp URL: https://www.iij.ad.jp/en/