2008 年の Debian OpenSSL インシデントにみるオープンソースソフトウェアのセキュリティ分析

西村　健†　　佐藤　周行‡

†国立情報学研究所
101-8430 東京都千代田区一ツ橋 2-1-2
takeshi@nii.ac.jp

‡東京大学
113-8658 東京都文京区弥生 2-11-16
schuko@satolab.itc.u-tokyo.ac.jp

あらまし　近年オープンソースソフトウェアは様々な分野で使用されるようになってきた．しかし，オープンソースソフトウェアはその品質に対する保証がないという一面が軽視されがちである．

今回，セキュリティ分野で有名なオープンソースソフトウェアであるOpenSSLに対してDebianが行なった修正により引き起こされたセキュリティインシデントについて考察する．これは生成される鍵ペアが予測可能になるという脆弱性をもたらしたが，この影響範囲を明らかにするとともに，国立情報学研究所で行なっている大学内へサーバ証明書を発行するプロジェクトにおける実際の使用例をもとに，オープンソースソフトウェアを利用する側が考えておくべきリスクを分析する．

Security Analysis of Open Source Software from 2008 Debian OpenSSL Incident

Takeshi Nishimura†　　Hiroyuki Sato‡

†National Institute of Informatics
2-1-2 Hitotsubashi Chiyoda-ku Tokyo, 101-8430 Japan
takeshi@nii.ac.jp

‡the University of Tokyo
2-11-16 Yayoi Bunkyo-ku Tokyo, 113-8658 Japan
schuko@satolab.itc.u-tokyo.ac.jp

Abstract Open source software is proved to be very useful in saving time and cost in building software of complex functions. Security is not an exception of this trend. A problem in securityware is the guarantee of its quality on security. In this paper, we analyze 2008 Debian incident on OpenSSL. The vulnerability on pseudo-random number generation is identified apart from the announcement of Debian. Furthermore, we have made an experiment on about 7,200,000,000 predictable key generations, and analyzed the usage of vulnerable keys in a certification authority.

1　はじめに

オープンソースソフトウェアは複雑な機能を持つソフトウェアの開発において，開発期間短縮やコスト削減に有効な手段であると認識されるようになってきた．オープンソースソフトウェアの潮流は，ライブラリや各種ツール，オペレーティングシステムを提供するまでに成長している．これ

らを利用して各種ソリューションのミドルウェアとしてオープンソースソフトウェアを使用するということも一般的になってきた.

セキュリティ分野にも同様にオープンソースソフトウェアの流れがあり, OpenSSL[7] や OpenSSH[6]はその代表例である.

セキュリティに関わるソフトウェアにおける関心事はその品質保証である. オープンソースソフトウェアの開発者がその品質に責任を持つことは当然であるが, 一方で何らかのオープンソースソフトウェアをミドルウェアとして使用するシステムを開発・運用する者は, そのミドルウェアの品質に責任を持たなければならない, ということは軽視されがちである.

つまり, オープンソースソフトウェアを含むシステムを運用する者は, そこに含まれる全てのオープンソースソフトウェアの品質を常に確かめる必要があるということである. 今日多くのシステム運用管理者はこの内部のオープンソースソフトウェアに対する継続的な品質管理を軽視しているようである.

2008 年 5 月 13 日に, Linux オペレーティングシステムの有名なディストリビュータの一つである Debian[5]は, Debian 内部で行なった OpenSSL に対する改変がもたらした脆弱性[1,8]について公開した. OpenSSL は認証局や電子証明書発行のミドルウェアとして広く利用されており, この脆弱性は深刻なものと受け止められた.

日本では, 国立情報学研究所(NII)が 2007 年よりパブリックなサーバ証明書発行を行なうプロジェクト[9]を実施している. このプロジェクトでは鍵ペアや CSR の生成において OpenSSL を用いてサーバ管理者側で生成することを前提としている. この点において NII はシステムのセキュアな運用のために OpenSSL の品質を継続的にチェックする責任を負っている.

OpenSSL は現在でも多くの人に信頼され高く評価されているソフトウェアであり, NII が OpenSSL を使用させると決定したことは妥当だと考えられる. ただし一旦 OpenSSL に関わるインシデントが発生すると, そのインシデントを詳細に調査する必要性が生じる.

この論文では, 上記インシデントの調査からその影響範囲を調べている. その結果新たに脆弱な鍵ペア群が判明したが, それらは深刻なものではないと判断される. また, NII のサーバ証明書プロジェクトにおいてこのインシデントがどのような影響を与えたかを調べ, これを通してオープンソースソフトウェアの利用者がこのようなインシデントに対してどのように対処すべきかを考察する.

2 Debian OpenSSL インシデントについて

2008 年 5 月 13 日に, Debian Security Advisory 1571(DSA 1571)という形で脆弱性の報告が行なわれた. それによれば, Debian が行なった OpenSSL の改変は疑似乱数生成ルーチンにおいてそれを推測可能とする脆弱性をもたらしたとされる[8]. 以下にその部分を引用する.

> The broken version of OpenSSL was being seeded only by process ID. Due to differences between endianness and sizeof(long), the output was architecture-specific: little-endian 32bit (e.g. i386), little-endian 64bit (e.g. amd64, ia64), big-endian 32bit (e.g. powerpc, sparc). PID 0 is the kernel and PID_MAX (32768) is not reached when wrapping, so there were 32767 possible random number streams per architecture. This is (2^15-1)*3 or 98301.

この中で, 疑似乱数のバリエーションが 98301 種しかないことが明示されており, 鍵長およびアルゴリズムを固定した場合 98301 個の鍵ペアが推測可能であり脆弱であるとみなされ

る. これに引き続き行われた他のグループによるアナウンス・解析においても同様の解釈がなされている.

この報告を受けて, 多くのセキュリティベンダや商用認証局が顧客に対して脆弱な Debian システムで鍵ペア生成を行なった場合は鍵ペアの更新および証明書の再発行を申請するよう, 注意喚起を行なっている.

NII についても例外ではなく, 広く注意喚起を行なうとともに, 前述の脆弱な鍵ペアを判定するツールを用いて既に発行した証明書全てをチェックし, 脆弱であると判定された証明書のサーバ管理者に個別に連絡を行なっている. 一方で多くのサーバ管理者は自身のサーバ環境がこのインシデントの脆弱なシステムに該当するかを調べ, 鍵ペア生成に問題のある OpenSSL を使用しなかったかをチェックしている. 結果鍵ペアに問題があると判断された場合には鍵ペアの再生成を行ない旧証明書の失効を行なった.

このインシデントに対する対応は, オープンソースソフトウェアの欠点を端的に表している. つまり, オープンソースソフトウェアである OpenSSL を利用する者は, インシデントに対して自ら何らかのアクションをとる必要があることを示している. これは, ベンダサポートのあるソフトウェアであれば契約に従ってベンダ側が何らかのアクションをとることと対極にある. もしベンダサポートのあるソフトウェアが内部でオープンソースソフトウェアを利用している場合でも, ベンダ側がアクションを起こさなければならないという点では同様である.

3 脆弱な鍵ペア群の拡張

3.1 推奨される OpenSSL の鍵ペア生成方法

OpenSSL はコマンドラインオプションとして -rand オプションを提供しており, これにより疑似乱数のエントロピーを増大させることができる. NII では以下のように 3 つのファイルを指定してエントロピーを増大させることを推奨している.

```
% openssl genrsa –rand file1:file2:file3 1024
```

OpenSSL は前述の環境による差異に加えて指定されたファイルの内容を種とすることにより, 乱数生成におけるエントロピーを増大させている.

3.2 –rand オプションにおけるインシデントによる影響

Debian からのアドバイザリおよび他のグループによる注意喚起に, -rand オプションを使用した場合を記述しているものは皆無であった. しかし我々の調査によればこのインシデントは -rand オプションを使用した場合にも少なからぬ影響を与えている.

Debian での OpenSSL の改変は, 乱数生成ルーチンへエントロピーを与える部分に対して行なわれ, 特定のバッファの内容を一切エントロピーに寄与させないというものであった. つまり -rand オプションを用いた場合でも例外ではなく, 指定されたファイルの内容は無視され, 指定されたファイルのサイズ[1]およびファイル数のみがエントロピーに寄与することになる. つまり前述の 98301 種の環境の差異に加えて-rand に指定するファイル数およびファイルサイズを固定すれば, 改変された OpenSSL は他の要因によらず一意の鍵ペアを生成することを意味する.

このような鍵ペアは予測可能という意味で脆弱と分類すべきであり, この節ではこのような鍵ペアを分析する.

3.3 鍵ペア生成のリプレイ

-rand オプションを用いた場合に生成される鍵ペアの脆弱性を調べる上で, そこでどの程度情報が欠落しているかを調べるのは重要である. 我々はパラメータを指定して実際に多くの鍵ペ

[1] 正確にはサイズを 1024 で割った商によって疑似乱数が特定される

アを生成することにより，それを確認することにした.

より具体的に述べると，OpenSSL での乱数生成のエントロピー増加に寄与するのは次のパラメータである. プロセス ID，乱数生成デバイス(/dev/urandom)，CPU のアーキテクチャ，-rand で指定するファイルサイズおよびその内容. このうち改変された OpenSSL では乱数生成デバイスの内容とファイル内容が無視される. 改変された OpenSSL に対して与えるパラメータを表 1 にまとめる. 実際には指定するファイル数やファイルサイズは無限であるが，NII での推奨値，および実際に指定されるファイルはすでにシステム上に存在するものを流用する傾向にあることから上限を設定した. また big-endian 32bit は利用例が少ないことから対象から除外した.

表 1 のパラメータで生成される鍵ペアの数は $32767 \times (33^3 + 33^2 + 33 + 1) \times 3 \times 2$,でおよそ 7G 個となる. それぞれパラメータでの鍵ペアは独立に計算できるので，並列計算機等を用いて計算を行なった.

結果は前述の判定ツールでのフォーマットに合わせて鍵ペアの SHA-1 ハッシュをとり，後半 10 バイトを格納している. これにより鍵ペアの統計的な情報を得ることはできなくなっているが，少なくともハッシュ値の一致を調べることから異なるパラメータから同一の鍵が生成されるかどうかを確認することはでき，結果，全てのパラメータから別々の鍵ペアが生成されることを確認した. すなわち，この鍵空間がパラメータ空間以上に小さくなることはなく，鍵ペアを予測するためには鍵生成に使用したものと全く同一のパラメータでリプレイを行なっていなければならないことを示している.

4 NII サーバ証明書プロジェクトでのインシデントの影響

前述の通り NII のサーバ証明書プロジェクトではこのインシデント発生から間もなく周知活動や個別の通知などを行なったが，この節では NII での発行・失効履歴から脆弱な鍵ペアがどれほど存在していたのかを調査する.

4.1 NII サーバ証明書プロジェクトについて

国立情報学研究所(NII)は、2007 年 5 月に大学等のサーバ証明書の普及推進と証明書発行プロセスの研究をすることを目的として「サーバ証明書の発行・導入における啓発・評価研究プロジェクト」[9]を開始した。NII サーバ証明書プロジェクトは WebTrust for CA 認定の主要なブラウザから信頼される商用認証局をルート認証局とするサーバ証明書を参加学術機関（大学等）に対して配付している。

2009 年 6 月から「UPKI オープンドメイン証明書自動発行検証プロジェクト」と呼ばれる新プロジェクト[10]が始まり，旧プロジェクトは 2009 年 9 月末に終了した. ここでは旧プロジェクトで発行した全2298枚のサーバ証明書を対象に解析を行なう.

4.2 脆弱な鍵ペアの出現

NII サーバ証明書プロジェクトにおいて発行した証明書の数およびその内訳を表 2 に示

表 1: 鍵ペア生成リプレイに用いるパラメータ

PID	1-32767
ファイルサイズ	0-32k 1k 刻み
ファイル数	0 or 1 or 2 or 3
.rnd	readable / unreadable / non-existent
CPU アーキテクチャ	32bit / 64bit little-endian

す．脆弱な鍵ペアの比率としては，世の中の状況をよく反映したものとなっている[3]．一な鍵ペアは存在するものの数は少なく，現実的な脅威としてはそれほど深刻なものではないと解釈される．

4.3 分析

表 3 に脆弱と判断された証明書の月別発行数を示す．

まず，脆弱性が公開された 2008 年 5 月以降に脆弱な Debian システムを利用して鍵生成された脆弱な証明書が 16 件発行されていることが分かる．そのうち 3 件は数カ月のうちに脆弱性のない鍵ペアによって更新されているものの，残り 13 件は本調査後の通知まで脆弱な証明書を使用し続けていた．13 件はその後の調査で 1CD Linux である KNOPPIX の脆弱性のあるバージョンを使用したことが分かっており，利用者が脆弱性に注意することはもちろん，認証局側でも継続的にチェックする必要性が示されている．

また，表 3 からインシデント発生時点で脆弱な証明書は 12 件あり，そのうち 4 件は -rand オプションが指定されていたために当初の脆弱性チェックでは検知できず，これらの管理者へは個別の注意喚起が行なわれなかったことが分かる．

表 4 は NII サーバ証明書プロジェクトにおいて失効理由を「Key Compromise」，つまり鍵の危殆化として失効申請された件数を示す．2009 年 7 月以降の 13 件は，前述のようにこちらからの通知に応じて失効されたものである．

本研究で脆弱だと判断した 28 件のうち正しく失効申請が出ていないものが 4 件ある．このうち 3 件は前述の通り更新されたものであるが，残り 1 件は個別通知が行なわれなかった 4 件のうちの 1 件であった．これは自身が当初の脆弱性チェックツールで「脆弱性なし」と判定されたために対処しなかったものと考えることもでき，正しくない脆弱性チェックツールが有害であることを示す可能性がある．

また，表 4 で 2008 年 5 月およびそれ以降に危殆化での失効件数が急増しており，これ

表 2: NII サーバ証明書プロジェクトでの脆弱な鍵ペアの出現

2298	発行された総証明書数
24	-rand オプションなしの脆弱な証明書数
4	-rand オプションありの脆弱な証明書数
1 0 3	ファイル数が 1 の脆弱な証明書数 ファイル数が 2 の脆弱な証明書数 ファイル数が 3 の脆弱な証明書数

表 3: 脆弱な証明書の月別発行数

発行月	発行枚数
2007 年 7 月	1 (-rand #1)
10 月	1
12 月	6
2008 年 2 月	1
4 月	3 (-rand #3)
2009 年 1 月	1
2009 年 2 月	15
計	28

(-rand #N のあるものは-rand オプションで N 個のファイルを指定した鍵ペア生成)

表 4: 月別の危殆化による失効数

失効月	失効枚数
2008 年 4 月	2
5 月	8 (5)
6 月	11 (6)
8 月	1
2009 年 7 月	2 (2)
8 月	11 (11)
計	35 (24)

(括弧内は本研究で脆弱と判断された証明書に対する失効の数)

らは本研究で判定できなかったが本質的に脆弱な証明書である可能性がある．最大で 9 件の脆弱な証明書をチェックできていない可能性があり，パラメータの上限を上げて鍵ペアを再生成してみるなど，確認を行なう必要がある．

5　関連研究

[1]は Debian OpenSSL インシデントの問題点を詳細に解析し，回避策を提示する．ただし-rand オプションを使用した脆弱な鍵ペアの拡張には触れられていない．

脆弱性を解析し分類する研究は多くあるが，中でも Mell ら[2]は CVSS という脆弱性評価システムを構築している．我々の研究は個々の事例から脆弱性をより深く解析するものである．

6　まとめ

この論文では，2008 年に起きた Debian OpenSSL インシデントの解析を行ない，どのような鍵ペアが脆弱であり予測可能になるかを精査した．結果従来言われていたよりも多くの鍵ペアが予測可能であることが分かったが，その鍵空間はパラメータ空間より小さくなることはなく，現実世界での使用頻度も少ないため深刻な問題ではないと結論付けた．

また NII サーバ証明書プロジェクトでの脆弱な鍵ペアの出現頻度から，オープンソースソフトウェアの利用者は常にその品質に目を光らせ，利用を促す者は継続的に啓蒙活動を行なわなければならないことが明らかとなった．

7　謝辞

この研究における計算の一部は東京大学情報基盤センターT2K オープンスーパーコンピュータ(HITACHI HA8000 クラスタシステム)を使用して行なった。

また，この研究に関して非常に有益な助言をいただいた国立情報学研究所島岡政基客員准教授に感謝いたします．

参考文献

[1] Ahmad, D., “Two Years of Broken Crypto: Debian’s Dress Rehearsal for a Global PKI Compromise”, IEEE Security & Privacy, Volume 6, Issue 5, pp.70-73, 2008.

[2] Mell, P., Scarfone, K., Romanosky, S., “Common Vulnerability Scoring System”, Security & Privacy, IEEE, Volume 4, Issue 6, pp.85-89, 2006.

[3] Nishimura, T., Sato, H., “Analysis of a Security Incident of Open Source Middleware － Case Analysis of 2008 Debian Incident of OpenSSL –”, Workshop on Middleware Architecture in International Symposium on Applications and the Internet 2009, pp.247-250, 2009.

[4] The American Institute of Certified Public Accountants, “WebTrust Program for Certification Authorities”,
http://www.webtrust.org/

[5] Debian – The Universal Operating System,
http://www.debian.org/

[6] OpenBSD Project, OpenSSH,
http://www.openssh.com/

[7] OpenSSL: The Open Source toolkit for SSL/TLS,
http://www.openssl.org/

[8] SSLkeys – Debian Wiki
http://wiki.debian.org/SSLkeys

[9] 国立情報学研究所, サーバ証明書の発行・導入における啓発・評価研究プロジェクト,
https://upki-portal.nii.ac.jp/docs/server

[10] 国立情報学研究所, UPKI オープンドメイン証明書自動発行検証プロジェクト,
https://upki-portal.nii.ac.jp/docs/odcert