Network Working Group
Request for Comments: 1948
Category: Informational

S. Bellovin
AT&T Research

May 1996

Защита от атак с подменой порядковых номеров

Defending Against Sequence Number Attacks

Статус документа

Этот документ является информационным и не содержит каких-либо стандартов Internet. Документ можно распространять без ограничений.

Тезисы

Атаки с подменой порядковых номеров TCP стали реальностью и представляют достаточно серьезную опасность (CERT Advisory CA-95:01[1]). Не отвергая средств криптографической аутентификации, автор документа предлагает внести простые изменения в существующие реализации протокола TCP, которые существенно усложнят организацию атак с подменой порядковых номеров.

Обзор

В 1985 году Моррис [1] описал атаку, основанную на предсказании порядковых номеров при организации новых соединений TCP [2]. Уязвимость протокола и некоторых реализаций позволяет организовать атаку на сервер с использованием в качестве адреса отправителя IP-адреса доверенного хоста и полностью выполнить 3-этапную процедуру согласования при организации соединений TCP с использованием предсказанных порядковых номеров. Для получения дополнительных сведений о порядковых номерах, используемых атакуемым хостом, с последним также организуется реальное соединение. После организации обманного соединения атакующий может выполнять команды на атакуемом сервере фактически от имени доверенного хоста.

Очевидно, что решением проблемы может служить криптографическая аутентификация [3,4], однако повсеместное развертывание такой системы потребует значительного времени. Следовательно, до решения проблемы многие хосты, использующие доверительные отношения и аутентификацию на основе адресов для таких приложений, как rlogin и rsh, должны будут ограничить использование этих приложений. К несчастью широкое распространение программ подслушивания и анализа пакетов (CERT Advisory CA-94:01[2]) делает использование протокола TELNET [5] весьма опасным. Таким образом, сеть Internet остается без надежного и безопасного механизма удаленного доступа (login).

В этом документе предлагаются простые изменения реализаций протокола TCP, которые позволят блокировать большинство атак с предсказанием порядковых номеров. Атаки не удастся предотвратить совсем, но выполнить их будет значительно сложней.

Описание атаки

Для понимания сути атак с предсказанием порядковых номеров рассмотрим процедуру 3-этапного согласования при организации соединений TCP [2]. Предположим, что хост A хочет подключиться к серверу rsh на хосте B. В этом случае передается пакет:

```
A->B: SYN, ISNa
```

Хост A передает пакет с флагом SYN (синхронизация порядковых номеров) и стартовым порядковым номером ISNa. Хост B передает в ответ

```
B->A: SYN, ISNb, ACK(ISNa)
```

В дополнение к передаче своего стартового порядкового номера хост B подтверждает полученный от A порядковый номер. Хост A завершает процедуру организации соединения, передавая пакет с подтверждением начального порядкового номера хоста B

```
A->B: ACK(ISNb)
```

Стартовые порядковые номера выбираются более или менее случайно. Точнее говоря, RFC 793 указывает, что 32-битовое значение счетчика стартовых номеров должно увеличиваться на 1 каждые 4 микросекунды. Вместо этого ядра, построенные на основе Berkeley, увеличивают счетчик каждую секунду но постоянное значение[3], а также используют дополнительное увеличение стартового номера на другую константу[4] при организации каждого соединения. Таким образом, организовав соединение с машиной, вы можете с высокой точностью предсказать стартовый порядковый номер для следующего соединения. Остальное – дело техники.

Атакующий хост X сначала создает реальное соединение с объектом атаки B (скажем с портом электронной почты или TCP echo). Это дает атакующему порядковый номер ISNb. Затем атакующий хост начинает передавать пакеты от имени подставного хоста A

```
Ax->B: SYN, ISNx
```

---

[1] См. http://www.cert.org/advisories/CA-1995-01.html. *Прим. перев.*
[2] См. http://www.cert.org/advisories/CA-1994-01.html. *Прим. перев.*
[3] Обычно 128. *Прим. перев.*
[4] Обычно на 64. *Прим. перев.*

Ax используется для обозначения пакетов от хоста X, который хочет прикинуться хостом A. Хост B ответит на полученный от X обманный запрос SYN пакетом

```
B->A: SYN, ISNb', ACK(ISNx)
```

адресованным легитимному хосту A, о котором мы поговорим чуть позже. X не увидит этот пакет, но передаст атакуемому серверу пакет

```
Ax->B: ACK(ISNb')
```

в котором содержится предсказанный порядковый номер ISNb. Если предсказание верно (а обычно это так) сервер rsh на хосте B будет думать, что с ним соединился легитимный хост A, хотя пакеты фактически передает хост X. Откликов от сервера хост X не будет получать, но ему и не нужны отклики. Задачей атаки является подключение к серверу для выполнения на нем команды и эта задача успешно решена атакующим.

На самом деле имеется одна дополнительная трудность. Если хост A получит пакет от атакуемого хоста B, он решит, что B подтверждает получение пакета, который ему не передавался и отправит в ответ сообщение с флагом RST, которое приведет к разрыву соединения. Существует несколько способов обхода этой сложности. Можно дождаться, пока хост A не утратит работоспособность (например, в результате атаки на него). Однако существует более простой и реальный способ “заглушить” хост A, основанный на весьма распространенной ошибке реализации протокола TCP, которая описана ниже.

Решение проблемы

Стартовые номера для соединений выбираются не случайно. Вместо этого используются алгоритмы, минимизирующие вероятность восприятия старых порядковых номеров для реинкарнации соединения [6, Appendix A]. Более того, реализации TCP, созданные на базе 4.2BSD, содержат специальный код, который обрабатывает ситуации, когда серверная сторона соединения остается в состоянии TIMEWAIT [7, стр. 945] при организации нового соединения. Простой метод использования случайных номеров, предложенный в статье [8], в таких случаях не будет работать.

Однако вопросы дубликатов и, следовательно, ограничение на стартовые номера для реинкарнаций соединения имеют отношение только к данному соединению. Т. е., между порядковыми номерами для различных соединений отсутствует синтаксическая или семантическая связь. Можно предотвратить предсказание порядковых номеров и связанные с этим атаки путем использования отдельного пространства порядковых номеров для каждого квартета из адресов и номеров портов отправителя и получателя. Внутри такого подпространства порядковые номера увеличиваются согласно алгоритму [2], однако между стартовыми номерами в каждом из подпространств не существует каких-либо предсказуемых связей.

Обычный способ реализации такого решения состоит в поддержке информации о состоянии “умерших” соединений и простейшим вариантом является изменение диаграммы состояний TCP таким образом, чтобы обе стороны всех соединений переходили в состояние TIMEWAIT. Этот способ будет работать, но потребует большого расхода ресурсов для хранения информации о состояниях. Взамен этого предлагается использование предложенного в стандарте 4-микросекундного таймера M и установка стартового номера с помощью функции

```
ISN = M + F(localhost, localport, remotehost, remoteport).
```

Важно обеспечить такую функцию F, значение которой невозможно рассчитать вне системы, и тогда атакующий не сможет предсказывать стартовые номера на основе порядковых номеров других соединений. Предлагается использовать в качестве F криптографическую хэш-функцию от идентификатора соединения и неких секретных данных. Хорошим решением может послужить алгоритм MD5 [9], поскольку код реализации этого алгоритма легко доступен. В качестве секретных данных может использоваться случайное значение [10] или комбинация специфического для хоста секрета и времени загрузки операционной системы на хосте. Время загрузки включено сюда для внесения элемента случайности. Можно включать в хэш-функцию и другие данные (адрес и имя хоста и т. п.) - это упростит администрирование, позволяя использовать общий секрет для всех хостов сети и разделяя пространство порядковых номеров за счет внесения дополнительных параметров. Рекомендуется на практике использовать все три элемента - случайные числа от доступного в системе генератора, административно задаваемый ключ (секрет) и IP-адреса хостов. Это существенно осложнит предсказание порядковых номеров для каждого хоста сети.

Отметим, что секретный ключ не удастся просто заменить на работающем компьютере. При замене ключа будут меняться стартовые порядковые номера для реинкарнаций соединения. Для безопасной замены ключа требуется сохранение информации об “умерших” соединениях или отсутствие активности в течение двух сроков жизни сегмента TCP.

Ошибка TCP

Как было отмечено выше при атаках с использованием предсказанных порядковых номеров требуется сначала “заглушить” доверенный хост, адрес которого будет использоваться в качестве подставного. Решить эту задачу можно разными способами, но в большинстве реальных атак использовалась ошибка реализации протокола.

При получении пакетов SYN с запросом на соединение сервер создает новую структуру TCB в состоянии SYN-RCVD. Чтобы избежать перерасхода ресурсов системы, построенные на основе 4.2BSD, поддерживают для каждого соединения ограниченное число TCB в состоянии SYN-RCVD. После достижения заданного порога новые пакеты SYN будут отбрасываться в предположении, что инициатор соединения передаст запрос повторно.

При получении пакета сначала следует проверить наличие TCB для данного соединения. При отсутствии TCB ядро просматривает "шаблоны"[1] TCB, используемые серверами для восприятия соединений от всех клиентов. К несчастью во многих вариантах ядра этот код используется для всех входящих пакетов, а не только для стартовых пакетов SYN. Если очередь SYN-RCVD для “шаблонных” TCB заполнена, все новые пакеты для данного хоста и номера порта будут отбрасываться, даже если они не содержат флага SYN.

Для того, чтобы заглушить хост атакующему достаточно передать несколько десятков пакетов SYN в порт rlogin с различными номерами портов отправителя от некого несуществующего хоста. Это приведет к заполнению очереди SYN-RCVD и пакеты SYN+ACK будут отбрасываться по причине переполнения. Атака в этом случае может быть

[1] В оригинале – wild card TCB. *Прим. перев.*

организована как обращения из порта rlogin доверенного хоста. Отклики (SYN+ACK) от атакуемого сервера будут приходить в переполненную очередь и отбрасываться. Этого можно избежать, если проверять состояние флага ACK, который не может быть установлен в пакетах, относящихся к легитимным соединениям. При обнаружении такого флага в ответ должен передаваться пакет с флагом RST.

Вопросы безопасности

Случайный выбор стартовых порядковых номеров не заменит криптографической аутентификации. В лучшем случае он обеспечит половинчатое решение проблемы.

Подслушивающий хост, который может видеть стартовые порядковые номера для соединений, способен определить состояние порядковых номеров и организовать атаку путем перехвата соединений. Однако, злоумышленник может также захватывать и существующие соединения [11]. Поскольку смещение между обманным и реальным соединением остается более или менее постоянным в течение времени жизни секретного ключа, важно не дать злоумышленнику возможности перехвата таких пакетов. Типичные атаки которые могут быть организованы на основе сказанного, будут включать подслушивание и различные атаки, описанные в статье [8].

Если случайные числа используются в качестве единственного секрета, они **должны** соответствовать рекомендациям [10].

Благодарности

Мэтт Блейз (Matt Blaze) и Джим Эллис (Jim Ellis) за ценные идеи.

Фрэнк Кастенхольц (Frank Kastenholz) за конструктивные замечания.

Литература

[1] R.T. Morris, "A Weakness in the 4.2BSD UNIX TCP/IP Software[1]", CSTR 117, 1985, AT&T Bell Laboratories, Murray Hill, NJ.

[2] Postel, J., "Transmission Control Protocol", STD 7, RFC 793[2], September 1981.

[3] Kohl, J., and C. Neuman, "The Kerberos Network Authentication Service (V5)", RFC 1510, September 1993.

[4] Atkinson, R., "Security Architecture for the Internet Protocol", RFC 1825[3], August 1995.

[5] Postel, J., and J. Reynolds, "Telnet Protocol Specification", STD 8, RFC 854, May 1983.

[6] Jacobson, V., Braden, R., and L. Zhang, "TCP Extension for High-Speed Paths", RFC 1885, October 1990.

[7] G.R. Wright, W. R. Stevens, "TCP/IP Illustrated, Volume 2", 1995. Addison-Wesley.

[8] S. Bellovin, "Security Problems in the TCP/IP Protocol Suite[4]", April 1989, Computer Communications Review, vol. 19, no. 2, pp. 32-48.

[9] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321[5], April 1992.

[10] Eastlake, D., Crocker, S., and J. Schiller, "Randomness Recommendations for Security", RFC 1750, December 1994.

[11] L. Joncheray, "A Simple Active Attack Against TCP"[6], 1995, Proc. Fifth Usenix UNIX Security Symposium.

Адрес автора

Steven M. Bellovin

AT&T Research

600 Mountain Avenue

Murray Hill, NJ 07974

телефон: (908) 582-5886

E-mail: smb@research.att.com

Перевод на русский язык

Николай Малых

nmalykh@gmail.com

[1]Перевод этой статьи вы можете найти на сайте www.protocols.ru *Прим. перев.*
[2]Перевод спецификации протокола TCP вы сможете найти на сайте www.protocols.ru *Прим. перев.*
[3]Этот документ заменен RFC 2401, который в последствии был заменен RFC 4301. *Прим. перев.*
[4]См. http://www.ja.net/CERT/Bellovin/TCP-IP_Security_Problems.html *Прим. перев.*
[5]Перевод этого документа имеется на сайте http://www.protocols.ru. *Прим. перев.*
[6]Перевод этой статьи вы можете найти на сайте www.protocols.ru *Прим. перев.*