БЕЗОПАСНЫЙ СЕРФИНГ В ИНТЕРНЕТ

В.П. Петлинский, А.В. Полещук

В настоящее время одними из основных угроз компьютерной безопасности является угрозы, связанные с серфингом в интернет. Фигурально выражаясь пользователи сами заражают свой компьютер,посещая небезопасные ресурсы глобальной сети. Примером тому может служить последняя эпидемия сетевого червя Conficker/Kido(модификации A,B,C), создавшего ботнет порядка 10 млн. компьютеров. Все возможные варианты угроз можно разделить на три основные группы. К первой можно отнести вирусы в обычном понимании, выполняющие деструктивные действия на компьютере. В этой группе существует своя внутренняя классификацмя, на которой мы останавливаться не будем, т.к. она уже была освещена достаточно подробно во многих источниках. Ко второй группе относится более широкий круг программ, получмвший название malware. Malware - вредная программа, т.е. программа, созданная со злым умыслом и/или злыми намерениями. Синонимы malware - badware, computer contaminant, crimeware. Сюда можно отнести шпионские программы, программы для финансовых преступлений и т.п. Подробная информация об этой группе программ и сетевых ресурсах, занимающихся их распространением представлена на сайте - http://www.stopbadware.org. Третья группа - это ресурсы сети, занятые фишингом (phishing) или интернет-мошенничеством. Технология интернет-мошенничества заключается в краже личных конфиденциальных данных, таких как пароли доступа,данные банковских и идентификационных карт и т.п. Количество фишинг сайтов составляет десятки тысяч. По мнению экспертов APWG (Anti-Phishing Working Group), это вызвано появлением так называемых phishing kit - утилит, позволяющих в короткие сроки создать фишинг-сайт. Появились и новые разновидности фишинга - вишинг (vishing), фарминг (pharming) и др. Более подробная информация о фишинге предоставлена на сайте APWG - http://www.antiphishing.org

В этих условиях необходимо использовать соответствующие средства защиты. Одним из основных средств защиты от угроз из интернет является проксирование. Технология проксирования в широком смысле означает взаимодействие через посредника (proxy). Наиболее распространенным является проксирование на прикладном уровне стека протоколов TCP/IP. В этом случае функции посредника возлагаются на прокси-сервер, поддерживающий набор протоколов прикладного уровня необходимый пользователям для взаимодействия с внешним миром. Основная функция посредника, защитить пользователя от угроз внешнего мира. Кроме того проксирование позволяет улучшить и ряд других характеристик взаимодействия с инернет, например, ускорять взаимодействие и/или уменьшить объём потреблямого трафика за счёт хранения актуальных данных в локальном кэше прокси-сервера. Для реализации задачи безопасного серфинга пользователей корпоративной сети ГНЦ РФ НИИАР в интернет нами используется иерархия прокси-серверов, приведенная на Рис.1. На жаргоне такая иерархия получила название сэндвич [1]. Прокси-серверы фунуционируют в составе двухузлового кластера высокой доступности, собранного в НИИАР, особенности реализации приведены в [2], Для балансировки запросов пользователей между узлами кластера, используется алгоритм CARP в скрипте автоопределения прокси на стороне клиента, описанный в [3].

Для повышения уровня доступности прокси-серверы в цепочке сконфигурированы таким образом, что могут обслуживать запросы нижестоящего уровня иерархии и для соседнего узла кластера (на рисунке помечены пунктиром).

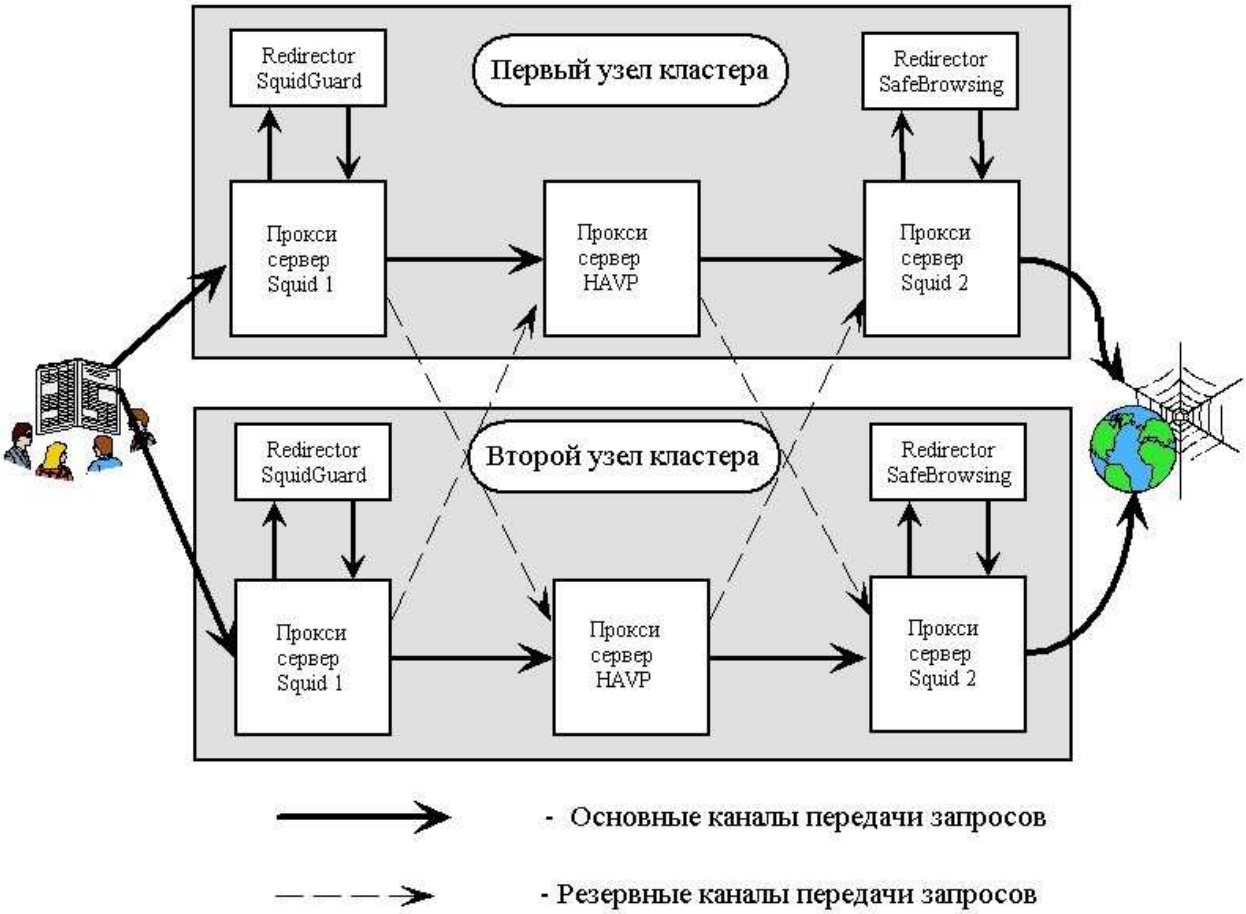

Рис.1 Иерархическая схема проксирования.

Рассмотрим более подробно функции каждого из прокси-серверов в цепочке. Первый прокси-сервер (Squid 1) на основе Squid Cache(Version 2.7.STABLE5) [4] выполняет функции аутентификации пользователей и ограничения доступа по ACL на основе их IP. Функция кэширования у него отключена по причине, описанной далее. Запрос поступает для анвлиза редиректору SquidGuard [5], который проверяет необходимость блокировки запросоа по его URL из баз данных блокировок. В НИИАР используется около 20 категорий блокировок, начиная от порно и азартных игр до нелицензионного ПО и шпионских сайтов (spyware). Например, в базе блокировок порно около 800000 URL, а в базе spyware около 30000 URL. В случае нахождения URL запроса в базах блокировок выполняется перенаправление (redirect) запроса на страницу диагностики, поясняющую причину блокировки. Этот же механизм используется нами и для борьбы с сетевым червём Conficker/Kido. Как известно, сам по себе на первом этапе он не представляет особой угрозы, занимаясь только построением ботнет-сети. Однако, в его функционал входит получение исполняемого деструктивного кода из интернет и запуск в действие по сигналу извне от управляющего центра. Доменные имена сайтов, с которых должен быть скачан деструктивный код, червь вычисляет по известному алгоритму в зависимости от текущей даты. Количество таких сайтов для модификаций A и B составляет — 250, а для модификации C - 50000(каждый день - новые). Мы используем программу, для вычисления этих имен, на каждый новый день и обновляем их в базе данных блокировок SquidGuard. Естественно, всё это выполняется автоматизированным способом. Авторы программы из Института Компьютерных Наук в Бонне [6] дали также блестящее описание функционала червя Conficker, намного превосходящее аналогичные от известных коммерческих антивирусных компаний, а также ряд полезных утилит для борьбы с ним.

Второй прокси сервер в цепочке - HAVP занимается только антивирусной проверкой на лету (авторами использована аббревиатура Http AntiVirusProxy - HAVP Version 0.90). Для проверки в HAVP используется функционал библиотеки libclamav широко известного антивируса ClamAV [7] с открытым исходным кодом, а также база антивирусных сигнатур от ClamAV. обновляемую стандартным образом через freshclam, входящий в дистрибутив ClamAV. Нами рассматривались три возможных варианта для антивирусной проверки - через протокол c-icap. через редиректор и наконец через специализированный прокси сервер. Последний вариант был выбран нами из-за отсутствия ограничений по масштабируемости и возможности многовендорной проверки.

Отметим, что при обслуживании 600 пользователей прокси и трафике около 3 Гбт в рабочее время суток, у нас запущено по 300 child процессов HAVP на каждом узле. При этом не ощущается задержек при доступе к интернет даже при пиковых нагрузках, т.е. антивирусная проверка действительно выполняется на лету. Если говорить о многовендорности, то в HAVP включена возможность подключения около 10 антивирусов,

включая AVG Socket Scanner, NOD32 Socket Scanner, DrWeb Socket Scanner и др. Приведём также количество обнружений вирусов за текущий год: январь - 172, февраль - 70, март - 28, апрель - 46, май - 31.

Последний прокси в цепочке (Squid 2) на основе Squid Cache, является кэширующим. Кэширование должно выполняться, до проверки вхолящего трафика на HAVP прокси, чтобы данные забираемые из кэша проверялись на свежих антивирусных сигнатурах. Кроме того запросы к прокси Squid 2 отправляются на проверку редиректору SafeBrowsing. Попутно отметим, что Squid позволяет обращаться только к одному редиректору, поэтому мы вынуждены его использовать именно здесь, хотя по смыслу он выполняет функции подобные первому редиректору. Здесь выполняется проверка URL запросов на блокировку в основном для второй и третьей групп угроз - malware и phishing сайтов. В качестве исходных данных для проверки используются данные, поставляемые поисковиком Google. В рамках проекта Google SafeBrowsing API компания предоставляет открытый доступ к API для работы с базой вредоносных сайтов, формируемой поисковиком в процессе анализа сайтов интернет. Сама база обновляется freshclam при включенной опции SafeBrowsing в конфигурационном файле. Поскольку эта опция появилась только в конце марта текущего года, то у нас пока недостаточно данных для оценки эффективности этого механизма защиты. Следует отметить, что опция SafeBrowsing на базе тех же API присутствует в ряде браузеров, например, Mozilla FireFox. Однако, проверка централизованным образом - на прокси представляется нам, более надёжной, эффективной и правомочной. Во-первых, эта опция присутствует не во всех браузерах, да и мало кто из пользователей ею пользуются. Во-вторых, это тормозит запрос, т.к. он до обработки отправляется в центр проверки Google. В-третьих, в этом случае может нарушаться privacy клиента, в случаях, когда на проверку отправляется запросс конфиденциальными данными.

К недостаткам отнесём, то что HAVP не может выполнять проверку шифрованных данных, по SSL соединению, что представляет определенную уязвимость в реализованном нами механизме защиты. Поэтому для ликвидации этого пробела, нами используется специальный ACL для работы по методу CONNECT протокола HTTP только с доверенными доменами. Ниже приведен этот фрагмент файла конфигурации Squid; GOOD_SSL_DOMAIN dstdomain "/etc/squid/GOOD_SSL_DOMAIN"

```
acl CONNECT method CONNECT
http_access deny CONNECT !SSL_ports
http_access deny CONNECT !GOOD_SSL_DOMAIN
```

В файле GOOD_SSL_DOMAIN указаны имена доверенных доменов.

Одно из основных преимуществ в реализованном нами механизме безопасного серфинга заключается в применении исключительно инструментов с открытым исходным кодом. При этом основной плюс мы видим не только в бесплатности, но также в свободе и гибкостив процессе дальнейшего развития этого механизма. В качестве примера можно отметить, например то, что все антивирусные продуты в основном используют сигнатурные методы для выявления вредоносного кода. Это означает, что выявление вредоносного ПО носит прецендентный характер. Если такое ПО создано исключительно против вашей компании, стандартные средства попросту его не обнаружат. В этой связи возникает потребность использовать защитные средства, учитывающие индивидуальные характеристики поведения клиента и соответствущие реакции на них. Мы полагаем, что такие средства достаточно просто могут интегрироваться в нашу систему в будущем.

ЛИТЕРАТУРА:

1. http://www.server-side.de/ - Домашняя страница разработчиков антивирусного прокси сервера HAVP
2. Петлинский В.П., Кинский А.О. Кластер простой архитектуры для поддержки Intranet/Internet сервисов ГНЦ РФ НИИАР на платформе Linux. В сборнике трудов 3-й Всероссийской научной конференции "Научный сервис в сети Ин-тернет", Новороссийск, 2001, Изд-во "Открытые системы".
3. Петлинский В.П. Способы распределения нагрузки в кластере Internet сервисов В сборнике трудов 4-й Всероссийской научной конференции "Научный сервис в сети Интернет", Новороссийск,2002, Изд-во МГУ.
4. http://www.squid-cache.org/ - Домашняя страница разработчиков прокси сервера Squid Cache
5. http://www.squidguard.org/ - Домашняя страница разработчиков редиректора SquidGuard
6. http://iv.cs.uni-bonn.de/wg/cs/applications/containing-conficker/ - Страница рабочей группы по борьбе с червём Conficker Института Компьютерных Наук в Бонне
7. http://www.clamav.org - Домашняя страница разработчиков антивируса ClamAV