

Shedding Zmiy: атака на Bitrix через уязвимость в виртуальной машине

rt-solar.ru/solar-4rays/blog/6220

January 1, 2001



В декабре прошлого года в ходе анализа инструментария проукраинской группировки Shedding Zmiy мы наткнулись на необычную утилиту. Она называлась Install и была кратко упомянута в [отчете](#) коллег из Positive Technologies об арсенале группировки. Мы детально проанализировали утилиту и выяснили, что в случае успешного получения несанкционированного доступа к веб-серверу она предназначалась для расширения привилегий в инфраструктуре через уязвимость (нулевого дня на тот момент) в настройках веб-сервера, включенного в состав инструмента «виртуальная машина», используемого для быстрого развертывания популярной российской CMS Bitrix. Об уязвимости мы сообщили вендору, и он ее оперативно закрыл. Эксплуатация возможна только на не обновленных версиях решения (до 9.0.5). Если вы еще не обновились, скорее сделайте это! Уязвимость включена в БДУ ФСТЭК — [BDU:2025-04539](#).

В этой статье расскажем о том, как атакующие эксплуатировали небезопасную настройку для реализации атаки и что еще могут сделать защитники организаций, чтобы затруднить реализацию атаки через системное окружение веб-приложений.

Утилита Install

Утилита Install — это инструмент, позволявший злоумышленнику повысить привилегии до root в результате эксплуатации учетной записи bitrix с расширенными правами доступа на веб-сервере, который обеспечивает работу ПО CMS 1C Bitrix.

Недостатки настроек в виртуальной машине для Bitrix

Суть проблемы заключалась в том, что на веб-сервере Apache административная учетная запись bitrix получала возможность модифицировать конфигурационные файлы Apache, что в дальнейшем могло привести к исполнению произвольного кода с повышенными привилегиями.

Условия эксплуатации уязвимости:

- Получение несанкционированного удаленного доступа к веб-серверу и размещение зловредной утилиты Install для установки зловредного модуля Apache.
- Администратор системы должен выполнить перезагрузку или запуск службы httpd, что активировало внесенные злоумышленником изменения.
- В случае если прямой доступ к перезапуску службы не был возможен, атакующему требовалось дожидаться ротации логов Apache, так как этот процесс также может привести к применению измененной конфигурации.

Также мы обнаружили и другие проблемы, связанные с расширенными правами административной учетной записи bitrix в системе. Все они являлись результатом повышенных привилегий для учетной записи.

- Обнаружена аналогичная проблема, связанная с nginx.
- Небезопасные расширенные настройки прав позволяли редактировать файлы службы push-service.

Цепочка эксплуатации

Install на вход получала путь до модуля Apache. По умолчанию использовался /usr/lib64/httpd/modules/mod_alias.so.

Утилита проверяла:

- префикс mod_.
- расширение .so.
- версию Apache.

MD5	848faa5839487c4331cb2a1146811f23
SHA1	f67dbe68fc11139b719fec11784247c5f6e7ea93
SHA256	37affeab7fb06a052413e9cc9272ea9cb2fd160fd204b506620d4303b06298c4
File name	install

File size 112.17 KB (114864 bytes)

Далее версия донора копировалась во вредоносный модуль, а затем генерировалось имя файла библиотеки. В качестве имени модуля утилита указывала test_module и создавала его в директории запуска. Если создание происходило успешно, то утилита выводила следующую инструкцию:

```
alex@pc:~/Documents/install_tool$ ./install
[+] Apache 2.4
[+] installing to ./mod_test.so
Add the following line to Apache config:

LoadModule test_module /home/alex/Documents/install_tool/mod_test.so

Restart apache, by
sudo -S /opt/webdir/bin/bx-site --email ...
Run
/usr/local/games/w 'sh -i'
```

Вывод Install после создания модуля

mod_test.so

MD5 2a30eac6fcf32dbf01de8af4280a7a06

SHA1 eab236f7f0df822b3df6b3920000992878d1ec63

SHA256 ee305286f2e3b35b9b596eedfef617a8f02dea04c8d58deaa4f8c1109194ec4e

File name mod_test.so

File size 34.36 kB

```

int start()
{
    int result; // eax
    int v1; // eax
    int v2; // ebp

    result = getuid();
    if ( !result )
    {
        v1 = open(file, 577, 511LL);
        v2 = v1;
        if ( v1 >= 0 )
            write(v1, buf, 0x6EA8uLL);
        close(v2);
        chown(file, 0, 0);
        return chmod(file, 04755u);
    }
    return result;
}

```

Код модуля

Данный модуль предназначен для генерации SUID-бинарного файла. Его основная и единственная функция — start, которая выполняла следующие действия:

- Создавала исполняемый файл по пути /usr/local/games/w.
- Устанавливала для данного файла права доступа 4755.

Права доступа 4755 включали в себя:

- SUID-бит (setuid), обеспечивающий выполнение файла с привилегиями владельца.
- Стандартные права 755 (владелец — чтение/запись/исполнение, группа и остальные — чтение/исполнение).

Таким образом, модуль позволял развернуть предварительно подготовленный исполняемый файл с повышенными привилегиями для дальнейшей эксплуатации.

/usr/local/games/w

SUID-бинарный файл, упакованный UPX, повышал привилегии через setuid и setgid:

```
setuid(0); // Установка root-привилегий для пользователя.  
setgid(0); // Установка root-привилегий для группы.
```

MD5	54b0634f06142261727aa3f60d87c2bf
SHA1	1335bf11489bc6321f384766fc2c00a339431918
SHA256	5e488e5dd25533130ecd4b4e220f9605d31fdb5d4af3293dfe004c38eb9fa0d0
File name	w
File size	56.66 KB

После этого он выполнял произвольную команду пользователя, переданную в кавычках. Например, на скриншоте с выводом в консоль можно было увидеть, что install предлагал передать в аргументы команду 'sh -i', что обеспечивало злоумышленнику shell с привилегиями суперпользователя. Также процесс перехватывал сигналы SIGQUIT и SIGINT.

Распределение прав

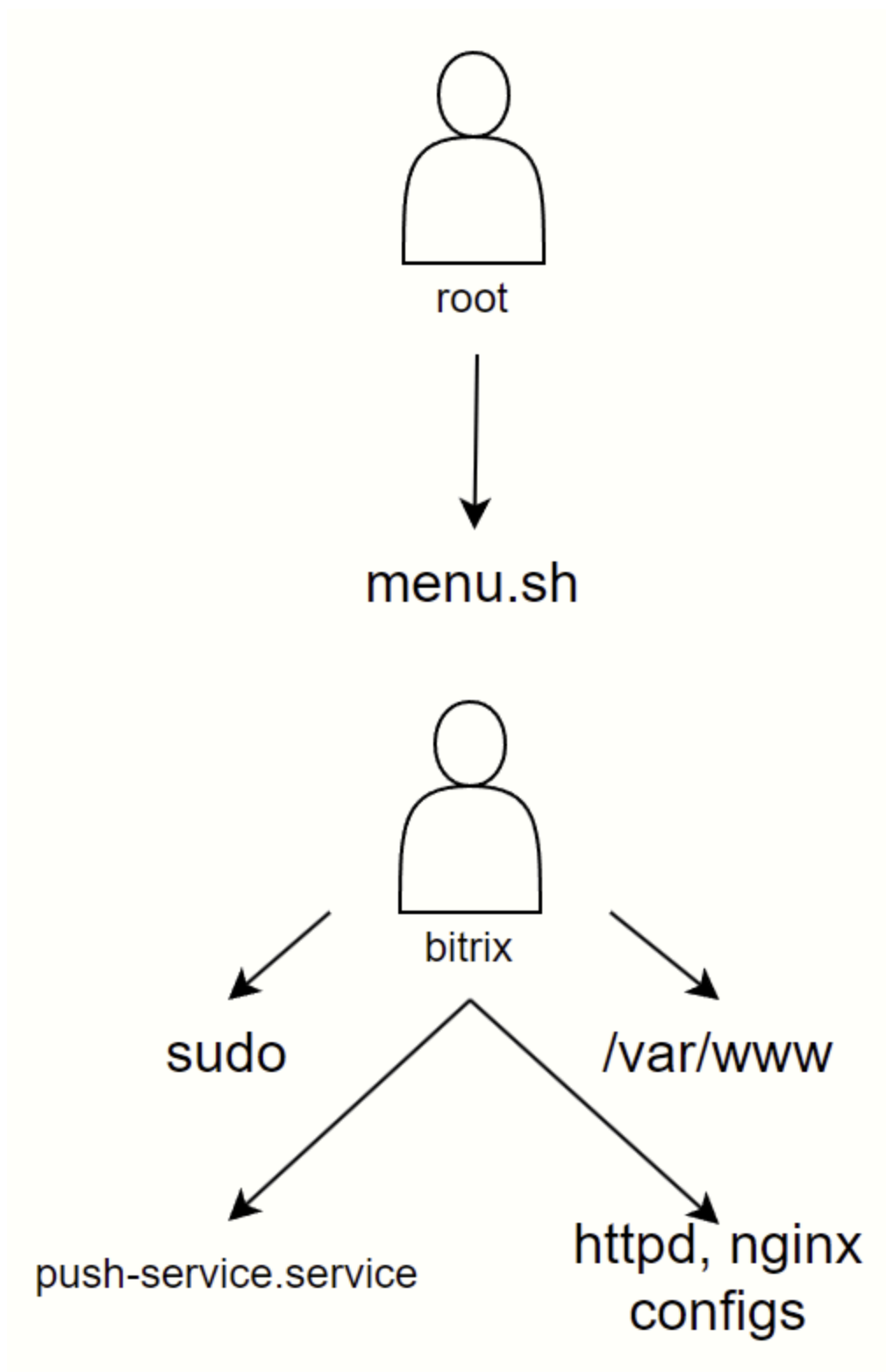
В виртуальной машине для развертывания ПО Bitrix есть административная учетная запись с правами, которые следует контролировать при построении защиты. В частности, в виртуальной машине устаревшей версии до 9.0.5 существовало два администратора, позволяющих управлять CMS. Настройка CMS осуществляется от пользователя root с помощью специального скрипта menu.sh. К обеим учетным записям имеется доступ по SSH.

В иерархии пользователей административная учетная запись bitrix занимала второе место после root и имела следующие права:

- владел директорией /var/www,
- доступ к sudo(хотя нет в sudoers),
- право редактировать файлы службы push-service,
- право редактировать конфигурацию httpd, nginx.

Права, выделенные красным цветом, позволяли злоумышленнику при получении доступа к административной учетной записи изменить настройки сервисов и в итоге получить права суперпользователя. Предусмотренные ошибочные настройки доступа к push-service

позволяли (пока они не были устранены в актуальных версиях виртуальной машины) напрямую внедрять код в службу. Данный подход к конфигурации нарушал принцип минимальных привилегий. Административной учетной записи bitrix было отдано слишком много функций по управлению системой.



Права редактирования файлов пользователей VM Bitrix устаревшей версии
9.0.0

Новейшие версии

В виртуальной машине версии 9.0.5 возможность доступа пользователя под административной учетной записью bitrix к конфигурации веб-серверов и файлам службы push-service убрали. А в новейшей доступной на сайте версии 9.0.6 домашней директорией bitrix стала /home/bitrix, где находится веб-контент и директория www.

```
bitrix 932 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 933 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 934 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 935 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 936 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 937 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 938 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 939 931 0 14:48 ? 00:00:00 nginx: worker process
bitrix 956 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 957 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 958 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 959 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 960 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 961 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 963 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 965 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 966 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
bitrix 970 901 0 14:48 ? 00:00:00 /usr/sbin/httpd -DFOREGROUND
```

Процессы Apache и Nginx работают от пользователя bitrix

Как усложнить злоумышленникам атаки через классические веб-приложения на примере виртуальной машины Bitrix

Первый и главный совет: следует своевременно устанавливать обновления. Это универсальный совет для любого ПО, в том числе для системного окружения к решениям Bitrix, и он помогает устранить большую часть рисков.

Дополнительно популярное ПО CMS и его системное окружение можно настроить так, чтобы максимально затруднить компрометацию. Такая настройка требует нескольких концептуальных и технических решений.

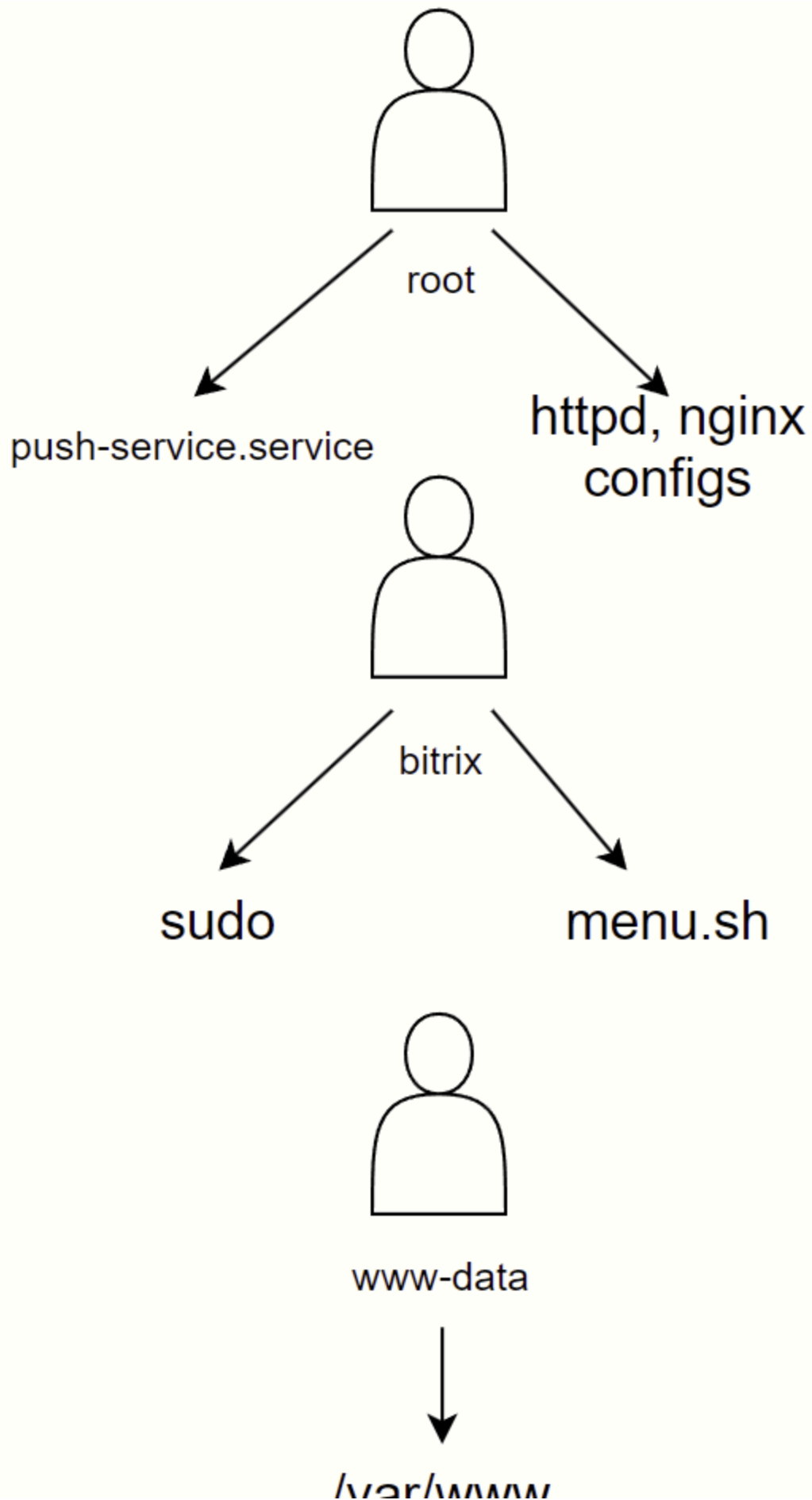
Самое главное решение — правильное разграничение прав пользователей.

Выделение отдельного пользователя

Почему эксплуатация обнаруженной нами уязвимости была возможна в Apache из состава виртуальной машины Bitrix, но невозможна в обычном веб-сервере Apache? Ответ заключается в организации доступов обычного Apache. Веб-сервер Apache использует дополнительного низко привилегированного пользователя www-data, который владеет директорией /var/www с файлами веб-сервера. Однако данный пользователь не имеет доступа к конфигурации Apache.

Поэтому важно правильно разграничить права пользователей.

В данном случае по аналогии с Apache напрашивается передача директории `www` системному пользователю `www-data` и запуск веб-серверов от него, чтобы минимизировать ущерб в случае взлома. Пользователь `www-data`, как правило, принадлежит к группе `www-data` и имеет минимальные права доступа к файлам и директориям, что делает систему менее уязвимой. Также необходимо оставить возможность доступа по SSH только у отдельной административной учетной записи, например `bitrix`.



Предлагаемая иерархия прав пользователей

С использованием такой конфигурации будет соблюдаться принцип наименьших привилегий. Администратор сможет управлять системой без необходимости исполнять каждую команду от суперпользователя. В то же время атакующие, попавшие в систему через эксплуатацию уязвимостей в фронте, не смогут сразу получить доступ к суперпользователю и аккаунту администратора.

Технические меры

Дополнительные меры безопасности могут быть введены с помощью Apparmor и SELinux. Этими средствами можно проактивно ограничить возможности атакующих в информационной системе, включающей веб-приложение. В частности, можно ограничить доступ к отдельным каталогам и исполнение отдельных утилит.

Так, для пользователя www-data должны быть оставлены доступы только для тех утилит, которые нужны для функционирования Apache и Nginx.

Для выявления аномального поведения при добавлении SUID\SGID-бита рекомендуется настроить мониторинг системных вызовов: chmod, fchmod, fchmodat. Отслеживая значения, передаваемые в их аргументах, возможно детектирование событий присвоения SUID-битов исполняемым файлам. Также возможен мониторинг исполнения системного вызова execve и с установленным битом SUID\SGID от пользователя, в частности www-data и bitrix. Это позволит детектировать попытки повышения привилегий в системе.

```
{
  "evt.arg.filename": "/usr/local/games/w",
  "evt.arg.mode": "S_IXOTH|S_IROTH|S_IXGRP|S_IRGRP|S_IXUSR|S_IWUSR|S_IRUSR|S_ISUID",
  "evt.args": "res=0 dirfd=-100(AT_FDCWD) filename=w(/usr/local/games/w)
mode=04755(S_IXOTH|S_IROTH|S_IXGRP|S_IRGRP|S_IXUSR|S_IWUSR|S_IRUSR|S_ISUID)",
  "evt.time": 1752570884440999448,
  "evt.type": "fchmodat"
}
```

Пример события установки SUID-бита на файл

Дополнительные меры

Важно понимать: даже если вы правильно разграничили права, наличие хотя бы одного скомпрометированного или уязвимого компонента в системном окружении CMS или каком-либо другом корпоративном ПО может нивелировать всю остальную работу по защите системы. Поэтому внедрение практик SCS (Supply Chain Security) и применение

специализированных инструментов анализа, таких как [Solar appScreener](#), может быть полезным дополнительным шагом в обеспечении устойчивости инфраструктуры. В сочетании с правильной настройкой прав доступа, использованием SELinux/AppArmor и регулярным аудитом это позволит выстроить действительно надежную модель защиты.

Следите за привилегиями

В арсеналах Shedding Zmiy и других группировок, атакующих российские организации, есть немало продвинутых инструментов, но нередко у злоумышленников нет необходимости к ним прибегать, поскольку эффективность показывают и менее изощренные техники атак. Это происходит потому, что в организациях порой не соблюдаются даже базовые меры безопасности, например правила распределения привилегий.

Правило наименьших привилегий является одним из основополагающих в вопросах обеспечения ИБ. Оно позволяет предотвратить быстрый захват сервера злоумышленником и дальнейшее его продвижение в инфраструктуре. Дополнительные технические меры защиты, а также мониторинг позволят предотвратить манипуляции атакующих и вовремя принять нужные меры. Это не единственная, но очень важная мера усиления защиты вашей инфраструктуры.

Примечание

Перед публикацией статьи мы поделились результатами нашего исследования с представителями «1С-Битрикс». Вот их комментарий:

Леонид Плетнев, бизнес-партнер по информационной безопасности «1С-Битрикс»

«Для удобства клиентов, мы выпускаем бесплатные решения, позволяющие быстро интегрировать наши бизнес-приложения в инфраструктуру заказчиков. В частности, «1С-Битрикс: Виртуальная машина» представляет собой свободно распространяемый сервер, адаптированный для оптимальной работы с любыми РНР-приложениями.

В рамках наших процессов выпуска безопасных продуктов «1С-Битрикс» активно привлекает независимую техническую экспертизу. В частности, мы довольны сотрудничеством с ГК «Солар» и отмечаем, что благодаря оперативному анализу зловредной утилиты специалистами компании мы смогли быстро ограничить предустановленные права доступа, выпустить обновление и свести на нет возможности несанкционированной эксплуатации одной из учетных записей веб-сервера.

Также отмечу, что обеспечение безопасности инфраструктуры, в которой работает бизнес ПО, — это взаимная задача, требующая вовлечения вендора, интеграторов, ИТ- и ИБ-служб клиентов. Для поддержания взаимного доверия прошу наших партнеров и клиентов уделять внимание аспектам безопасности системного окружения приложений,

пользоваться предоставляемыми нами возможностями по настройке защиты, оперативно устанавливая обновления безопасности, внимательно настраивать и контролировать уровни доступа».

IOC

MD5

848faa5839487c4331cb2a1146811f23
54b0634f06142261727aa3f60d87c2bf
2a30eac6fcf32dbf01de8af4280a7a06

SHA1

f67dbe68fc11139b719fec11784247c5f6e7ea93
eab236f7f0df822b3df6b3920000992878d1ec63
1335bf11489bc6321f384766fc2c00a339431918

SHA256

5e488e5dd25533130ecd4b4e220f9605d31fdb5d4af3293dfe004c38eb9fa0d0
ee305286f2e3b35b9b596eedfef617a8f02dea04c8d58deaa4f8c1109194ec4e
37affeab7fb06a052413e9cc9272ea9cb2fd160fd204b506620d4303b06298c4

Pandora Hive Mind

Содержание статьи

Получать новости

-
-

Сила киберразведки: знания об угрозах на страже защиты бизнеса

[Читать](#)