Solar
appScreener

Результаты анализа

1c.zip

Дата создания отчёта
08.06.2021 10:20:41

Автор отчёта
s.deev@rt-solar.ru

Способ классификации
ОУД4

Версия продукта
3.9.3

СОДЕРЖАНИЕ

ИНФОРМАЦИЯ О ПРОЕКТЕ

Название	1c.zip
UUID	ce3167e7-4272-418b-904c-3ead8029c262
Перейти к проекту в	appScreener

Динамика уровня безопасности

Уровень безопасности приложения оценивается по шкале от 0 (плохо) до 5 (отлично). Рейтинг вычисляется исходя из количества критических уязвимостей и уязвимостей среднего уровня. Влияние критических уязвимостей больше, чем влияние уязвимостей среднего уровня, и не учитывает объем кода. Уязвимости среднего уровня учитываются из расчёта их количества на общее число строк исходного кода.

Динамика количества уязвимостей

Уязвимости поделены на четыре категории: критические, среднего уровня, низкого уровня и информационного уровня.

1. **Критические уязвимости** с большой вероятностью приводят к компрометации конфиденциальных данных и нарушению целостности системы.

2. Уязвимости **среднего уровня** могут с меньшей вероятностью привести к компрометации конфиденциальных данных и нарушению целостности системы либо являются менее серьёзными нарушениями безопасности.

3. Уязвимости **низкого уровня** могут стать потенциальной угрозой безопасности.

4. Уязвимости **информационного уровня** сигнализируют о нарушении хороших практик программирования.

В первую очередь уделите внимание уязвимостям критического и среднего уровней.

История сканирований

Номер	Дата и время	Статус	Языки	Строки кода	Количество уязвимостей					Рейтинг
					Критический	Средний	Низкий	Инфо	Всего	
1/1	16.02.2021 21:47:15	завершено	1C	170	22	38	16	19	95	0.9/5.0

ПОДРОБНЕЕ ОБ ОУД4

Данный отчёт содержит информацию на соответствие ПО четвёртому оценочному уровню доверия (ОУД4).
«С 1 января 2020 г. в силу вступают новые положения Банка России, согласно которым финансовые организации будут обязаны проводить анализ уязвимостей прикладного ПО, которое используется для проведения платежных и других финансовых операций. При этом, ПО должно соответствовать оценочному уровню доверия (ОУД) не ниже четвертого – требования к уровням доверия описаны в ГОСТ 15408-3.»*

AVA_VAN.3.1D

Разработчик должен представить объект оценки для тестирования

AVA_VAN.3.1C

Объект оценки должен быть пригоден для тестирования

AVA_VAN.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств

AVA_VAN.3.2E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать потенциальные уязвимости в объекте оценки

AVA_VAN.3.3E

Оценщик должен провести независимый анализ уязвимостей объекта оценки с использованием документации руководств, функциональной спецификации, проекта объекта оценки, описания архитектуры безопасности и представления реализации, чтобы идентифицировать потенциальные уязвимости в объекте оценки

* П.9 «Положения Банка России от 17 апреля 2019 г. № 684-П “Об установлении обязательных для некредитных финансовых организаций требований к обеспечению защиты информации при осуществлении деятельности в сфере финансовых рынков в целях противодействия осуществлению незаконных финансовых операций”»

AVA_VAN.3.4E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы сделать заключение, что объект оценки является стойким к нападениям, выполняемым нарушителем, обладающим Усиленным базовым потенциалом нападения.

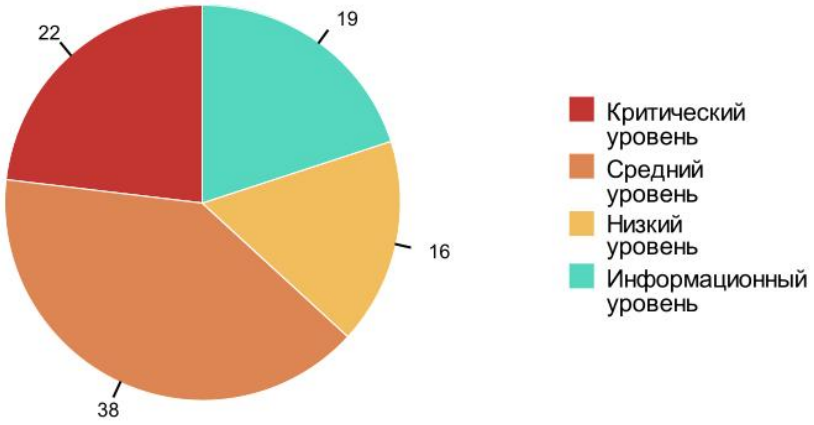
ИНФОРМАЦИЯ О СКАНИРОВАНИИ

1/1 16.02.2021 21:47:15

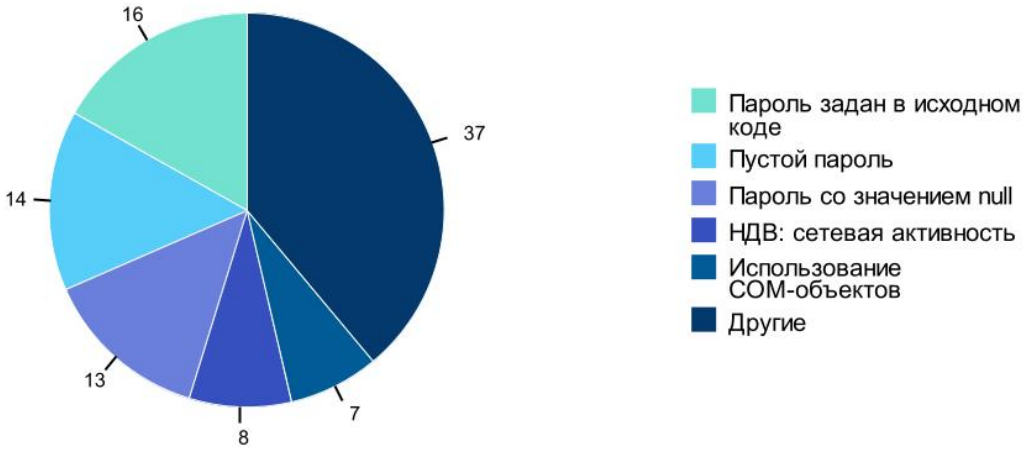
Статистика сканирования

Статус	завершено
Рейтинг	0.9/5.0
Продолжительность	0:00:00
Строки кода	170
Уязвимости	Критический 22 Средний 38 Низкий 16 Инфо 19 Всего 95

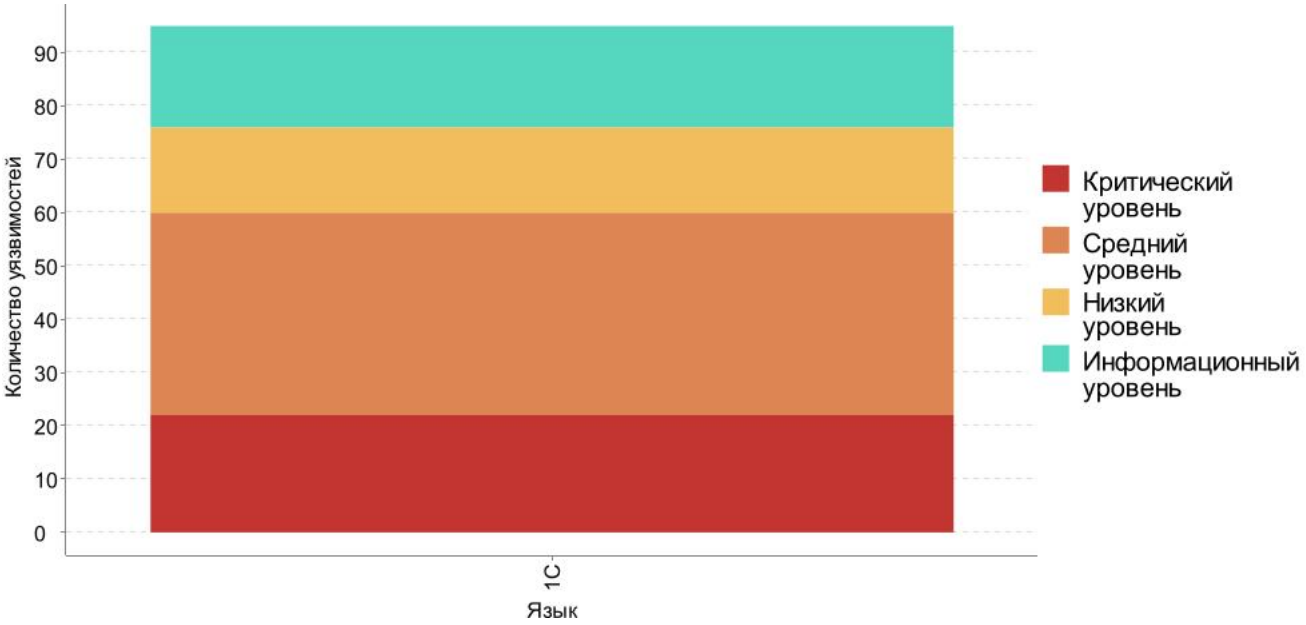
Найденные уязвимости



Типы уязвимостей



Статистика по языкам



Язык	Статус	Продолжительность	Строки кода	Количество уязвимостей				
				Критический	Средний	Низкий	Инфо	Всего
1C	завершено	0:00:00	170	22	38	16	19	95

Результаты тестирования на соответствие ОУД4

AVA_VAN.3.1D

Объект оценки предоставлен для тестирования

AVA_VAN.3.1C

Объект оценки пригоден для тестирования

AVA_VAN.3.1E

Представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств

AVA_VAN.3.2E

Уязвимости в библиотеках не обнаружены. Убедитесь в том, что при запуске анализа были выбраны опции Анализировать вложенные библиотеки и архивы и Анализировать стандартные библиотеки (JavaScript).

AVA_VAN.3.3E

Обнаружены уязвимости

Не обработано: 22 критического уровня, 38 среднего уровня

AVA_VAN.3.4E

Уязвимости с угрозой проникновения не обнаружены

Список уязвимостей

Уязвимости отображены с учётом настроек экспорта: отобрано **60**

Актуальных: **60** из **95**

AVA_VAN.3.3E		
Уязвимости критического уровня		22*
Пароль задан в исходном коде	1C	12
1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:2		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:2		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:4		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:6		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:10		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:12		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:14		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:16		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:18		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:22		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:24		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:30		Не обработано
Ключ шифрования задан в исходном коде	1C	1
1c/ONES_CRYPTO_KEY_HARDCODED.bsl:2		Не обработано
Слабый алгоритм шифрования	1C	2
1c/ONES_CRYPTO_BAD_ALGORITHM.bsl:2		Не обработано
1c/ONES_CRYPTO_BAD_ALGORITHM.bsl:6		Не обработано
Пустой пароль	1C	7

* Отклонённые уязвимости не учитываются

AVA_VAN.3.3E		
Уязвимости критического уровня		
Пустой пароль	1C	
1c/ONES_PASSWORD_EMPTY.bsl:2		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:4		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:10		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:14		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:16		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:18		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:30		Не обработано
Уязвимости среднего уровня		38*
НДВ: скрытая функциональность	1C	2
1c/ONES_BACKDOOR_HIDDEN_FUNCTIONALITY.bsl:2		Не обработано
1c/ONES_BACKDOOR_HIDDEN_FUNCTIONALITY.bsl:4		Не обработано
Пароль задан в исходном коде	1C	4
1c/ONES_PASSWORD_HARDCODED.bsl:8		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:20		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:27		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:43		Не обработано
Слабое зерно генератора псевдослучайных чисел	1C	1
1c/ONES_CRYPTO_BAD_SEED.bsl:2		Не обработано
Ключ шифрования задан в исходном коде	1C	1
1c/ONES_CRYPTO_KEY_HARDCODED.bsl:5		Не обработано

* Отклонённые уязвимости не учитываются

AVA_VAN.3.3E		
Уязвимости среднего уровня		
Слабый генератор псевдослучайных чисел	1C	
1c/ONES_CRYPTO_BAD_RANDOM.bsl:2		Не обработано
1c/ONES_CRYPTO_BAD_SEED.bsl:2		Не обработано
НДВ: специальная учётная запись	1C	5
1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:2		Не обработано
1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:2		Не обработано
1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:6		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:24		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:27		Не обработано
Использование незащищённого протокола HTTP	1C	6
1c/ONES_HTTP_USAGE.bsl:2		Не обработано
1c/ONES_HTTP_USAGE.bsl:17		Не обработано
1c/ONES_HTTP_USAGE.bsl:20		Не обработано
1c/ONES_HTTP_USAGE.bsl:23		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:35		Не обработано
1c/ONES_PASSWORD_HARDCODED.bsl:35		Не обработано
Небезопасные настройки политик cross-origin resource sharing	1C	1
1c/ONES_HTML5_CORS.bsl:3		Не обработано
Пустой пароль	1C	4
1c/ONES_PASSWORD_EMPTY.bsl:6		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:12		Не обработано
1c/ONES_PASSWORD_EMPTY.bsl:22		Не обработано

* Отклонённые уязвимости не учитываются

AVA_VAN.3.3E		
Уязвимости среднего уровня		
Пустой пароль	1C	
1c/ONES_PASSWORD_EMPTY.bsl:24		Не обработано
НДВ: сетевая активность	1C	5
1c/ONES_BACKDOOR_NETWORK_ACTIVITY.bsl:5#13		Не обработано
1c/ONES_HTTP_USAGE.bsl:2		Не обработано
1c/ONES_HTTP_USAGE.bsl:17		Не обработано
1c/ONES_HTTP_USAGE.bsl:20		Не обработано
1c/ONES_HTTP_USAGE.bsl:23		Не обработано
Пустой ключ шифрования	1C	2
1c/ONES_CRYPTO_KEY_EMPTY.bsl:2		Не обработано
1c/ONES_CRYPTO_KEY_EMPTY.bsl:4		Не обработано
Слабый алгоритм хеширования	1C	5
1c/ONES_CRYPTO_BAD_HASH.bsl:2		Не обработано
1c/ONES_CRYPTO_BAD_HASH.bsl:6		Не обработано
1c/ONES_CRYPTO_BAD_HASH.bsl:14		Не обработано
1c/ONES_CRYPTO_BAD_HASH.bsl:17		Не обработано
1c/ONES_CRYPTO_BAD_HASH.bsl:19		Не обработано
Уязвимости низкого уровня		0*
Уязвимости информационного уровня		0*

* Отклонённые уязвимости не учитываются

Подробные результаты

Ключ шифрования задан в исходном коде (1С)

AVA_VAN.3.3E

Описание

Ключ шифрования явно задан в исходном коде. Это может привести к компрометации данных приложения.
Устранить угрозы безопасности, связанные с заданными в исходном коде ключами, очень сложно. Такие ключи как минимум доступны каждому разработчику приложения. Более того, после того как приложение установлено, удалить из его кода ключ можно только посредством обновления. Константные строки легко извлекаются из скомпилированного приложения декомпиляторами. Поэтому злоумышленнику не обязательно иметь доступ к исходному коду, чтобы узнать значение используемого ключа.
Уязвимости типа «утечка конфиденциальных данных» (Sensitive Data Exposure) занимают третье место в рейтинге уязвимостей web-приложений «OWASP Top 10 2017».

Пример

В следующем примере используется ключ, который явно задан в коде приложения:
КлючШифрования = "hardcodedencryptionkey";

Рекомендации

- Не используйте заданные в исходном коде ключи.
- Для создания ключей рекомендуется использовать криптографически стойкий ГПСЧ (генератор псевдослучайных чисел).
- Для хранения ключей следует использовать специальные решения, такие как Аппаратные Модули Безопасности (Hardware Security Module).
- В целях безопасности необходимо периодически обновлять ключи.

Ссылки

Русскоязычные ссылки

1. Управление криптографическими ключами
2. Секретные ключи никогда не должны записываться в явном виде на носителе, который может быть считан или скопирован.

Англоязычные ссылки

1. OWASP: Use of hard-coded cryptographic key
2. CWE-321: Use of Hard-coded Cryptographic Key
3. OWASP Top 10 2013-A5-Security Misconfiguration

4. OWASP Top 10 2013-A6-Sensitive Data Exposure
5. OWASP Top 10 2017-A3-Sensitive Data Exposure
6. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration
7. CWE-798: Use of Hard-coded Credentials

Вхождения

1c/ONES_CRYPTO_KEY_HARDCODED.bsl:2

Уровень Критический

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_KEY_HARDCODED keyh03

2 КлючШифрования = "efsajfejrgkdfksdmflwe";

3
4 // <yes> <report> ONES_CRYPTO_KEY_HARDCODED keyh05
5 Функция СоединениеССервером(Хост, Логин = "admin", secretkey = "123456")
Экспорт
```

1c/ONES_CRYPTO_KEY_HARDCODED.bsl:5

Уровень Средний

Статус Не обработано

```
2 КлючШифрования = "efsajfejrgkdfksdmflwe";
3
4 // <yes> <report> ONES_CRYPTO_KEY_HARDCODED keyh05

5 Функция СоединениеССервером(Хост, Логин = "admin", secretkey = "123456")
Экспорт

6     //do smth
7 КонецФункции
```

Пароль задан в исходном коде (1С)

AVA_VAN.3.3E

Описание

Пароль явно задан в исходном коде. Это может привести к компрометации данных приложения.
Устранить угрозы безопасности, связанные с заданными в исходном коде паролями, очень сложно. Такие пароли как минимум доступны каждому разработчику приложения. Более того, после того как приложение установлено, удалить из его кода пароль можно только посредством обновления. Константные строки легко извлекаются из скомпилированного приложения декомпиляторами. Поэтому злоумышленнику не обязательно иметь доступ к исходному коду, чтобы узнать параметры специальной учётной записи. Если эти параметры станут известны злоумышленнику, администраторам системы придётся либо пренебречь безопасностью, либо ограничить доступ к приложению.

Пример

В следующем примере пароль задаётся в исходном коде приложения:
Пароль = "БольшойБольшойСекрет";

Рекомендации

- Храните не пароли, а значения криптографически стойкой хеш-функции от паролей. Используйте специализированные хеш-функции, предназначенные для этой цели (bcrypt, PBKDF2, scrypt). Используйте соль, полученную из криптографически стойкого генератора псевдослучайных чисел, для борьбы с атаками, использующими радужные таблицы.
- Если заданный в исходном коде пароль используется для первого входа в систему, предусмотрите для этой цели специальный режим аутентификации, при котором пользователь обязан предоставить собственный уникальный пароль.
- Храните аутентификационную информацию в зашифрованном виде в отдельном конфигурационном файле или в базе данных. Обеспечьте безопасную защиту ключа шифрования. Если использовать шифрование невозможно, максимально ограничьте доступ к хранилищу.

Ссылки

Англоязычные ссылки

1. Use of hard-coded password
2. CWE-259: Use of Hard-coded Password
3. OWASP Top 10 2013-A5-Security Misconfiguration
4. OWASP Top 10 2013-A6-Sensitive Data Exposure
5. Handling passwords used for auth in source code — stackoverflow.com
6. How to securely hash passwords? — security.stackexchange.com
7. OWASP Top 10 2017 A2-Broken Authentication
8. OWASP Top 10 2017-A3-Sensitive Data Exposure
9. CWE-798: Use of Hard-coded Credentials

10. CWE CATEGORY: OWASP Top Ten 2017 Category A2 - Broken Authentication
11. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration

Вхождения

1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:2

Уровень Критический

Статус Не обработано

```
1 // <yes> <report> ONES_BACKDOOR_SPECIAL_ACCOUNT acc001 <yes> <report>
ONES_PASSWORD_HARDCODED pash10


2 Если ИмяПользователя = "Администратор" И Пароль = "пароль" Тогда


3     ДатьНеограниченныйДоступ();
4 КонецЕсли
5 // <yes> <report> ONES_BACKDOOR_SPECIAL_ACCOUNT acc002
```

1c/ONES_PASSWORD_HARDCODED.bsl:2

Уровень Критический

Статус Не обработано

```
1 // <yes> <report> ONES_PASSWORD_HARDCODED pash01


2 СтрокаПодключения = "Srvr=""******"";Ref=""******"";Usr=""ExData"";Pwd=""secret"";";


3 // <yes> <report> ONES_PASSWORD_HARDCODED pash01
4 Connect = a.Connect("srvr = ""srv""; ref = ""Buh""; usr = ""Администратор""; pwd =
""secret"";");
5 // <yes> <report> ONES_PASSWORD_HARDCODED pash02
```

1c/ONES_PASSWORD_HARDCODED.bsl:4

Уровень Критический

Статус Не обработано

```
1 // <yes> <report> ONES_PASSWORD_HARDCODED pash01
2 СтрокаПодключения = "Srvr=""******"";Ref=""******"";Usr=""ExData"";Pwd=""secret"";";
3 // <yes> <report> ONES_PASSWORD_HARDCODED pash01
```

```
4 Connect = a.Connect("srvr = ""srv""; ref = ""Buh""; usr = ""Администратор""; pwd = ""secret"";");


5 // <yes> <report> ONES_PASSWORD_HARDCODED pash02
6 Пароль = "secret";
7 // <yes> <report> ONES_PASSWORD_HARDCODED pash03
```

1c/ONES_PASSWORD_HARDCODED.bsl:6

Уровень Критический

Статус Не обработано

```
3 // <yes> <report> ONES_PASSWORD_HARDCODED pash01
4 Connect = a.Connect("srvr = ""srv""; ref = ""Buh""; usr = ""Администратор""; pwd = ""secret"";");
5 // <yes> <report> ONES_PASSWORD_HARDCODED pash02


6 Пароль = "secret";


7 // <yes> <report> ONES_PASSWORD_HARDCODED pash03
8 Mypassword = "secret";
9 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report> ONES_BACKDOOR_NETWORK_ACTIVITY netw01
```

1c/ONES_PASSWORD_HARDCODED.bsl:10

Уровень Критический

Статус Не обработано

```
7 // <yes> <report> ONES_PASSWORD_HARDCODED pash03
8 Mypassword = "secret";
9 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report> ONES_BACKDOOR_NETWORK_ACTIVITY netw01


10 Ftp = Новый FTPСоединение("1.1.1.1", 21, "login", "secret");


11 // <yes> <report> ONES_PASSWORD_HARDCODED pash06
12 ПользовательAD.SetPassword("secret");
13 // <yes> <report> ONES_PASSWORD_HARDCODED pash01
```

1c/ONES_PASSWORD_HARDCODED.bsl:12

Уровень Критический

Статус Не обработано

```
9 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01
10 Ftp = Новый FTPСоединение("1.1.1.1", 21, "login", "secret");
11 // <yes> <report> ONES_PASSWORD_HARDCODED pash06

12 ПользовательAD.SetPassword("secret");

13 // <yes> <report> ONES_PASSWORD_HARDCODED pash01
14 АДОКоннектор.Open("Provider=ADsDSOObject;User Id="+Логин + ";Password=""
secret"";");
15 // <yes> <report> ONES_PASSWORD_HARDCODED pash05
```

1c/ONES_PASSWORD_HARDCODED.bsl:14

Уровень Критический

Статус Не обработано

```
11 // <yes> <report> ONES_PASSWORD_HARDCODED pash06
12 ПользовательAD.SetPassword("secret");
13 // <yes> <report> ONES_PASSWORD_HARDCODED pash01

14 АДОКоннектор.Open("Provider=ADsDSOObject;User Id="+Логин + ";Password=""
secret"";");

15 // <yes> <report> ONES_PASSWORD_HARDCODED pash05
16 ОбъектАД = Root_AD.OpenDSObject(ПутьОбъекта,Логин,"secret",100);
17 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw03
```

1c/ONES_PASSWORD_HARDCODED.bsl:16

Уровень Критический

Статус Не обработано

```
13 // <yes> <report> ONES_PASSWORD_HARDCODED pash01
14 АДОКоннектор.Open("Provider=ADsDSOObject;User Id="+Логин + ";Password=""
secret"";");
15 // <yes> <report> ONES_PASSWORD_HARDCODED pash05

16 ОбъектАД = Root_AD.OpenDSObject(ПутьОбъекта,Логин,"secret",100);

17 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw03
18 КлиентSSH = Новый КлиентSSH("127.0.0.1", 22, "user", "secret");
19 // <yes> <report> ONES_PASSWORD_HARDCODED pash08
```

1c/ONES_PASSWORD_HARDCODED.bsl:18

Уровень Критический

Статус Не обработано

```
15 // <yes> <report> ONES_PASSWORD_HARDCODED pash05
16 ОбъектАД = Root_AD.OpenDSObject(ПутьОбъекта,Логин,"secret",100);
17 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw03

18 КлиентSSH = Новый КлиентSSH("127.0.0.1", 22, "user", "secret");

19 // <yes> <report> ONES_PASSWORD_HARDCODED pash08
20 FTP.ПарольПрокси = "secret";
21 // <yes> <report> ONES_PASSWORD_HARDCODED pash07
```

1c/ONES_PASSWORD_HARDCODED.bsl:22

Уровень Критический

Статус Не обработано

```
19 // <yes> <report> ONES_PASSWORD_HARDCODED pash08
20 FTP.ПарольПрокси = "secret";
21 // <yes> <report> ONES_PASSWORD_HARDCODED pash07

22 FTP.Пароль = "secret";

23 // <yes> <report> ONES_PASSWORD_HARDCODED pash10 <yes> <report>
```

```
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001
24 Если Пароль = "secret" Тогда
25 КонецЕсли;
```

1c/ONES_PASSWORD_HARDCODED.bsl:24

Уровень Критический

Статус Не обработано

```
21 // <yes> <report> ONES_PASSWORD_HARDCODED pash07
22 FTP.Пароль = "secret";
23 // <yes> <report> ONES_PASSWORD_HARDCODED pash10 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001


24 Если Пароль = "secret" Тогда


25 КонецЕсли;
26 // <yes> <report> ONES_PASSWORD_HARDCODED pash09 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001
27 Если МойПароль = "secret" Тогда
```

1c/ONES_PASSWORD_HARDCODED.bsl:30

Уровень Критический

Статус Не обработано

```
27 Если МойПароль = "secret" Тогда
28 КонецЕсли;
29 // <yes> <report> ONES_PASSWORD_HARDCODED pash11


30 Функция СоединениеССервером(Хост, Логин = "admin", Пароль = "123456")
Экспорт


31   Соединение = Неопределено;
32   ЗаписатьВЖурналРегистрации("Соединение с сервером","Информация", , ,"
Попытка соединения с сервером");
33   Попытка
```

1c/ONES_PASSWORD_HARDCODED.bsl:8

Уровень Средний

Статус Не обработано

```
5 // <yes> <report> ONES_PASSWORD_HARDCODED pash02
6 Пароль = "secret";
7 // <yes> <report> ONES_PASSWORD_HARDCODED pash03


8 Mypassword = "secret";


9 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01
10 Ftp = Новый FTPСоединение("1.1.1.1", 21, "login", "secret");
11 // <yes> <report> ONES_PASSWORD_HARDCODED pash06
```

1c/ONES_PASSWORD_HARDCODED.bsl:20

Уровень Средний

Статус Не обработано

```
17 // <yes> <report> ONES_PASSWORD_HARDCODED pash04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw03
18 КлиентSSH = Новый КлиентSSH("127.0.0.1", 22, "user", "secret");
19 // <yes> <report> ONES_PASSWORD_HARDCODED pash08


20 FTP.ПарольПрокси = "secret";


21 // <yes> <report> ONES_PASSWORD_HARDCODED pash07
22 FTP.Пароль = "secret";
23 // <yes> <report> ONES_PASSWORD_HARDCODED pash10 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001
```

1c/ONES_PASSWORD_HARDCODED.bsl:27

Уровень Средний

Статус Не обработано

```
24 Если Пароль = "secret" Тогда
25 КонецЕсли;
26 // <yes> <report> ONES_PASSWORD_HARDCODED pash09 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001
```

```
27 Если МойПароль = "secret" Тогда


28 КонецЕсли;
29 // <yes> <report> ONES_PASSWORD_HARDCODED pash11
30 Функция СоединениеССервером(Хост, Логин = "admin", Пароль = "123456")
Экспорт
```

1c/ONES_PASSWORD_HARDCODED.bsl:43

Уровень Средний

Статус Не обработано

```
40   Возврат Соединение;
41 КонецФункции
42 // <yes> <report> ONES_PASSWORD_HARDCODED pash13


43 ЗаписьZipФайла(ПутьАрхива,"КлючШифрования",,,,МетодШифрованияZIP.
AES256);
```

Пустой пароль (1С)

AVA_VAN.3.3E

Описание

Пустой пароль может привести к компрометации приложения.
Устранить угрозы безопасности, связанные с заданными в исходном коде пустыми паролями, очень сложно. Информация о том, что определённая учётная запись принимает пустой пароль, как минимум доступна каждому разработчику приложения. Более того, после того как приложение установлено, удалить из его кода пустой пароль можно только посредством обновления. Константные строки легко извлекаются из скомпилированного приложения декомпиляторами. Поэтому злоумышленнику не обязательно иметь доступ к исходному коду, чтобы узнать параметры специальной учётной записи. Если эти параметры станут известны злоумышленнику, администраторам системы придётся либо пренебречь безопасностью, либо ограничить доступ к приложению.

Пример

Следующий код инициализирует переменную-пароль пустой строкой. Затем код пытается прочитать сохранённое значение пароля и сравнить его со значением, введённым пользователем.

```
    Перем ПарольПользователя;
```

```
...

Пароль = "";

Если !((ПарольПользователя = getPassword(Пароль)) = null) Тогда
    Пароль = ПарольПользователя;
КонецЕсли;

Если Пароль = userInput Тогда
    // Access private data
КонецЕсли
```

Если getPassword() не сможет получить пароль из базы данных, то злоумышленник сможет обойти проверку пароля, введя в качестве userInput пустую строку.

Рекомендации

- Не используйте пустые пароли.
- Храните не пароли, а значения криптографически стойкой хеш-функции от паролей. Используйте специализированные хеш-функции, предназначенные для этой цели (bcrypt, PBKDF2, scrypt). Используйте соль, полученную из криптографически стойкого генератора псевдослучайных чисел, для борьбы с атаками, использующими радужные таблицы.
- Если заданный в исходном коде пароль используется для первого входа в систему, предусмотрите для этой цели специальный режим аутентификации, при котором пользователь обязан предоставить собственный уникальный пароль.
- Храните аутентификационную информацию в зашифрованном виде в отдельном конфигурационном файле или в базе данных. Обеспечьте безопасную защиту ключа шифрования. Если использовать шифрование невозможно, максимально ограничьте доступ к хранилищу.

Ссылки

Англоязычные ссылки

1. Use of hard-coded password
2. CWE-259: Use of Hard-coded Password
3. OWASP Top 10 2013-A5-Security Misconfiguration
4. OWASP Top 10 2013-A6-Sensitive Data Exposure
5. Handling passwords used for auth in source code — stackoverflow.com
6. How to securely hash passwords? — security.stackexchange.com
7. OWASP Top 10 2017 A2-Broken Authentication
8. CWE CATEGORY: OWASP Top Ten 2017 Category A2 - Broken Authentication
9. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration

Вхождения

1c/ONES_PASSWORD_EMPTY.bsl:2

Уровень Критический

Статус Не обработано

```
1 // <yes> <report> ONES_PASSWORD_EMPTY pass01

2 СтрокаПодключения = "Srvr=""******"";Ref=""******"";Usr=""ExData"";Pwd="""";";

3 // <yes> <report> ONES_PASSWORD_EMPTY pass01
4 Connect = a.Connect("srvr = ""srv""; ref = ""Buh""; usr = ""Администратор""; pwd =
""""";");
5 // <yes> <report> ONES_PASSWORD_EMPTY pass02
```

1c/ONES_PASSWORD_EMPTY.bsl:4

Уровень Критический

Статус Не обработано

```
1 // <yes> <report> ONES_PASSWORD_EMPTY pass01
2 СтрокаПодключения = "Srvr=""******"";Ref=""******"";Usr=""ExData"";Pwd="""";";
3 // <yes> <report> ONES_PASSWORD_EMPTY pass01

4 Connect = a.Connect("srvr = ""srv""; ref = ""Buh""; usr = ""Администратор""; pwd =
""""";");

5 // <yes> <report> ONES_PASSWORD_EMPTY pass02
6 Пароль = "";
7 // <yes> <report> ONES_PASSWORD_EMPTY pass03
```

1c/ONES_PASSWORD_EMPTY.bsl:10

Уровень Критический

Статус Не обработано

```
7 // <yes> <report> ONES_PASSWORD_EMPTY pass03
8 Mypassword = "";
9 // <yes> <report> ONES_PASSWORD_EMPTY pass04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01

10 Ftp = Новый FTPСоединение("1.1.1.1", 21, "login", "");

11 // <yes> <report> ONES_PASSWORD_EMPTY pass06
12 ПользовательAD.SetPassword("");
13 // <yes> <report> ONES_PASSWORD_EMPTY pass01
```

1c/ONES_PASSWORD_EMPTY.bsl:14

Уровень Критический

Статус Не обработано

```
11 // <yes> <report> ONES_PASSWORD_EMPTY pass06
12 ПользовательAD.SetPassword("");
13 // <yes> <report> ONES_PASSWORD_EMPTY pass01

14 АДОКоннектор.Open("Provider=ADsDSOObject;User Id="+Логин + ";
Password="""";");

15 // <yes> <report> ONES_PASSWORD_EMPTY pass05
16 ОбъектАД = Root_AD.OpenDSObject(ПутьОбъекта,Логин,"",100);
17 // <yes> <report> ONES_PASSWORD_EMPTY pass04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw03
```

1c/ONES_PASSWORD_EMPTY.bsl:16

Уровень Критический

Статус Не обработано

```
13 // <yes> <report> ONES_PASSWORD_EMPTY pass01
14 АДОКоннектор.Open("Provider=ADsDSOObject;User Id="+Логин + ";
Password="""";");
15 // <yes> <report> ONES_PASSWORD_EMPTY pass05

16 ОбъектАД = Root_AD.OpenDSObject(ПутьОбъекта,Логин,"",100);
```

```
17 // <yes> <report> ONES_PASSWORD_EMPTY pass04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw03
18 КлиентSSH = Новый КлиентSSH("127.0.0.1", 22, "user", "");
19 // <yes> <report> ONES_PASSWORD_EMPTY pass08
```

1c/ONES_PASSWORD_EMPTY.bsl:18

Уровень Критический

Статус Не обработано

```
15 // <yes> <report> ONES_PASSWORD_EMPTY pass05
16 ОбъектАД = Root_AD.OpenDSObject(ПутьОбъекта,Логин,"",100);
17 // <yes> <report> ONES_PASSWORD_EMPTY pass04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw03

18 КлиентSSH = Новый КлиентSSH("127.0.0.1", 22, "user", "");

19 // <yes> <report> ONES_PASSWORD_EMPTY pass08
20 FTP.ПарольПрокси = "";
21 // <yes> <report> ONES_PASSWORD_EMPTY pass07
```

1c/ONES_PASSWORD_EMPTY.bsl:30

Уровень Критический

Статус Не обработано

```
27 Если МойПароль = "" Тогда
28 КонецЕсли;
29 // <yes> <report> ONES_PASSWORD_EMPTY pass12

30 Процедура СоединениеССервером(Хост, Логин = "admin", Пароль = "") Экспорт

31   Соединение = Неопределено;
32   ЗаписатьВЖурналРегистрации("Соединение с сервером","Информация", , ,"
Попытка соединения с сервером");
33   Попытка
```

1c/ONES_PASSWORD_EMPTY.bsl:6

Уровень Средний

Статус Не обработано

```
3 // <yes> <report> ONES_PASSWORD_EMPTY pass01
4 Connect = a.Connect("srvr = ""srv""; ref = ""Buh""; usr = ""Администратор""; pwd =
""""";");
5 // <yes> <report> ONES_PASSWORD_EMPTY pass02


6 Пароль = "";


7 // <yes> <report> ONES_PASSWORD_EMPTY pass03
8 Mypassword = "";
9 // <yes> <report> ONES_PASSWORD_EMPTY pass04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01
```

1c/ONES_PASSWORD_EMPTY.bsl:12

Уровень Средний

Статус Не обработано

```
9 // <yes> <report> ONES_PASSWORD_EMPTY pass04 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01
10 Ftp = Новый FTPСоединение("1.1.1.1", 21, "login", "");
11 // <yes> <report> ONES_PASSWORD_EMPTY pass06


12 ПользовательAD.SetPassword("");


13 // <yes> <report> ONES_PASSWORD_EMPTY pass01
14 АДОКоннектор.Open("Provider=ADsDSOObject;User Id="+Логин + ";
Password="""";");
15 // <yes> <report> ONES_PASSWORD_EMPTY pass05
```

1c/ONES_PASSWORD_EMPTY.bsl:22

Уровень Средний

Статус Не обработано

```
19 // <yes> <report> ONES_PASSWORD_EMPTY pass08
20 FTP.ПарольПрокси = "";
21 // <yes> <report> ONES_PASSWORD_EMPTY pass07


22 FTP.Пароль = "";


23 // <yes> <report> ONES_PASSWORD_EMPTY pass10 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001
24 Если Пароль = "" Тогда
25 КонецЕсли;
```

1c/ONES_PASSWORD_EMPTY.bsl:24

Уровень Средний

Статус Не обработано

```
21 // <yes> <report> ONES_PASSWORD_EMPTY pass07
22 FTP.Пароль = "";
23 // <yes> <report> ONES_PASSWORD_EMPTY pass10 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001


24 Если Пароль = "" Тогда


25 КонецЕсли;
26 // <yes> <report> ONES_PASSWORD_EMPTY pass09 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001
27 Если МойПароль = "" Тогда
```

Слабый алгоритм шифрования (1C)

AVA_VAN.3.3E

Описание

Приложение использует слабый алгоритм шифрования.
Устаревшие алгоритмы шифрования не обеспечивают достаточной защиты для приложений, работающих с ценными данными. Безопасность алгоритма шифрования определяется предполагаемыми затратами времени и ресурсов, необходимыми для получения доступа к зашифрованной информации. Разработка новых методов атак и увеличение вычислительной мощности компьютеров приводят к устареванию алгоритмов, ранее считавшихся безопасными. Например, DES из-за небольшой длины ключа (56 бит) может быть взломан методом полного перебора.
Для обеспечения защиты ценных данных следует использовать протестированные реализации стандартизированных алгоритмов шифрования с достаточной длиной

ключа.
Уязвимости типа «утечка конфиденциальных данных» (Sensitive Data Exposure) занимают третье место в рейтинге уязвимостей web-приложений «OWASP Top 10 2017».

Пример

В следующем примере показана инициализация шифрования по устаревшему алгоритму RC2:

```
objCrypt = Новый COMОбъект("System.Security.Cryptography.RC2");
```

Примеры уязвимых алгоритмов шифрования: RC2, RC4, DES. Безопасный вариант:

```
objCrypt = Новый COMОбъект("System.Security.Cryptography.AES");
```

Рекомендации

- Используйте современные стандартизированные алгоритмы шифрования (например, AES).
- Ознакомьтесь с рекомендациями специализированных организаций (OWASP, NIST) по вопросам рекомендованной длины ключа и других параметров шифрования.

Ссылки

Русскоязычные ссылки

1. Криптостойкость алгоритма DES
2. Взлом TLS и SSL через уязвимость в шифре RC4

Англоязычные ссылки

1. OWASP Top 10 2017-A3-Sensitive Data Exposure
2. OWASP Top 10 2013-A6-Sensitive Data Exposure
3. CWE-327
4. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration

Вхождения

1c/ONES_CRYPTO_BAD_ALGORITHM.bsl:2

Уровень Критический

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_BAD_ALGORITHM algo01 <yes> <report>
ONES_COM_USAGE com002

2 objCrypt = Новый COMОбъект("System.Security.Cryptography.RC2");

3 // <yes> <report> ONES_COM_USAGE com002
4 EncryptedData = Новый COMОбъект("CAPICOM.EncryptedData");
5 // <yes> <report> ONES_CRYPTO_BAD_ALGORITHM algo02
```

1c/ONES_CRYPTO_BAD_ALGORITHM.bsl:6

Уровень Критический

Статус Не обработано

```
3 // <yes> <report> ONES_COM_USAGE com002
4 EncryptedData = Новый COMОбъект("CAPICOM.EncryptedData");
5 // <yes> <report> ONES_CRYPTO_BAD_ALGORITHM algo02

6 EncryptedData.Algorithm.Name = CAPICOM_ENCRYPTION_ALGORITHM_RC2;
```

Использование незащищённого протокола HTTP (1C)

AVA_VAN.3.3E

Описание

Использование HTTP вместо HTTPS позволяет реализовать атаку «человек посередине». Это может привести к полной утрате конфиденциальности передаваемых данных.
Использование протокола HTTPS, основанного на HTTP и SSL / TLS, позволяет защитить передаваемые данные от несанкционированного доступа и изменения. Рекомендуется использовать HTTPS для всех случаев передачи ценной информации между клиентом и сервером, в частности, для страницы авторизации и всех страниц,

требующих аутентификации.

Пример

В следующем примере приложение осуществляет соединение по протоколу HTTP:
Соединение = Новый HTTPСоединение("example.org");

Рекомендации

- Используйте только безопасные протоколы (например, HTTPS) для передачи конфиденциальных данных между клиентом и сервером.

Ссылки

Русскоязычные ссылки

1. Чем отличается HTTP от HTTPS?
2. Мигрируем на HTTPS / habrahabr.ru

Англоязычные ссылки

1. OWASP Top 10 2017-A3-Sensitive Data Exposure
2. Transport Layer Protection Cheat Sheet – OWASP
3. Web Security: Why You Should Always Use HTTPS – Mike Shema / Mashable
4. CWE-319: Cleartext Transmission of Sensitive Information
5. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration

Вхождения

1c/ONES_HTTP_USAGE.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_HTTP_USAGE http01 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw02

2 Соединение = Новый HTTPСоединение("thumb7.shutterstock.com", 80);

3
4 // <no> <report>
5 Соединение = Новый HTTPСоединение(
```

1c/ONES_HTTP_USAGE.bsl:17

Уровень Средний

Статус Не обработано

```
14
15 Процедура ВыполнитьЗапрос(Команда)
16     // <yes> <report> ONES_HTTP_USAGE http02 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01

17     ВыполнитьHTTPЗапрос("http://example.com");

18 КонецПроцедуры
19 // <yes> <report> ONES_HTTP_USAGE http02 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01
20 HttpАдрес = "http://example.com";
```

1c/ONES_HTTP_USAGE.bsl:20

Уровень Средний

Статус Не обработано

```
17     ВыполнитьHTTPЗапрос("http://example.com");
18 КонецПроцедуры
19 // <yes> <report> ONES_HTTP_USAGE http02 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01

20 HttpАдрес = "http://example.com";

21
22 // <yes> <report> ONES_HTTP_USAGE http01 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw02
23 Соединение = Новый HTTPСоединение("www.mysite.ru");
```

1c/ONES_HTTP_USAGE.bsl:23

Уровень Средний

Статус Не обработано

```
20 HttpАдрес = "http://example.com";
21
22 // <yes> <report> ONES_HTTP_USAGE http01 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw02

23 Соединение = Новый HTTPСоединение("www.mysite.ru");
```

1c/ONES_PASSWORD_EMPTY.bsl:35

Уровень Средний

Статус Не обработано

```
32 ЗаписатьВЖурналРегистрации("Соединение с сервером","Информация", , ,"
Попытка соединения с сервером");
33 Попытка
34    // <yes> <report> ONES_HTTP_USAGE http01

35    Соединение = Новый HTTPСоединение(Хост, , Логин, Пароль, , Ложь);

36    ЗаписатьВЖурналРегистрации("Соединение с сервером","Информация", , ,"
Соединение успешно установлено!");
37 Исключение
```

```
38          ЗаписатьВЖурналРегистрации("Соединение с сервером","Ошибка", , ,
ОписаниеОшибки());
```

1c/ONES_PASSWORD_HARDCODED.bsl:35

Уровень Средний

Статус Не обработано

```
32 ЗаписатьВЖурналРегистрации("Соединение с сервером","Информация", , ,"
Попытка соединения с сервером");
33 Попытка
34    // <yes> <report> ONES_HTTP_USAGE http01

35    Соединение = Новый HTTPСоединение(Хост, , Логин, Пароль, , Ложь);

36    ЗаписатьВЖурналРегистрации("Соединение с сервером","Информация", , ,"
Соединение успешно установлено!");
37 Исключение
38          ЗаписатьВЖурналРегистрации("Соединение с сервером","Ошибка", , ,
ОписаниеОшибки());
```

НДВ: сетевая активность (1С)

AVA_VAN.3.3E

Описание

Приложение инициирует соединение с заданным в исходном коде внешним сервером. Если адрес не входит в список заведомо безопасных, это может свидетельствовать о недокументированной сетевой активности приложения.
Данная уязвимость может привести к утечке конфиденциальных данных. Согласно рейтингу уязвимостей web-приложений, утечка конфиденциальных данных (Sensitive Data Exposure) занимает третье место в «OWASP TOP 10 2017».

Пример

В следующем примере приложение соединяется с сервером по заданному IP-адресу:

```
Соединение = Новый HTTPСоединение(
    "79.174.66.120",
    443
);
```

Рекомендации

- Если адрес, с которым происходит соединение, не входит в список заведомо безопасных, удалите участки кода, использующие подозрительное соединение.

Ссылки

Русскоязычные ссылки

1. Википедия: Программная закладка

Англоязычные ссылки

1. The Art of the Backdoor
2. CWE-506: Embedded Malicious Code

Вхождения

1c/ONES_BACKDOOR_NETWORK_ACTIVITY.bsl:5#13

Уровень Средний

Статус Не обработано

```
2 ip = "1.1.12.23";
3
4 // <yes> <report> ONES_BACKDOOR_NETWORK_ACTIVITY netw02


5 Соединение = Новый HTTPСоединение(
6         "ya.ru", // сервер (хост)
7         443, // порт, по умолчанию для http используется 80, для https 443
8         , // пользователь для доступа к серверу (если он есть)
9 ...
10          , // здесь указывается прокси, если он есть
11          , // таймаут в секундах, 0 или пусто - не устанавливать
12        Новый ЗащищенноеСоединениеOpenSSL()
13     );
```

1c/ONES_HTTP_USAGE.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_HTTP_USAGE http01 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw02


2 Соединение = Новый HTTPСоединение("thumb7.shutterstock.com", 80);


3
4 // <no> <report>
5 Соединение = Новый HTTPСоединение(
```

1c/ONES_HTTP_USAGE.bsl:17

Уровень Средний

Статус Не обработано

```
14
15 Процедура ВыполнитьЗапрос(Команда)
16     // <yes> <report> ONES_HTTP_USAGE http02 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01

17     ВыполнитьHTTPЗапрос("http://example.com");

18 КонецПроцедуры
19 // <yes> <report> ONES_HTTP_USAGE http02 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01
20 HttpАдрес = "http://example.com";
```

1c/ONES_HTTP_USAGE.bsl:20

Уровень Средний

Статус Не обработано

```
17     ВыполнитьHTTPЗапрос("http://example.com");
18 КонецПроцедуры
19 // <yes> <report> ONES_HTTP_USAGE http02 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw01

20 HttpАдрес = "http://example.com";

21
22 // <yes> <report> ONES_HTTP_USAGE http01 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw02
23 Соединение = Новый HTTPСоединение("www.mysite.ru");
```

1c/ONES_HTTP_USAGE.bsl:23

Уровень Средний

Статус Не обработано

```
20 HttpАдрес = "http://example.com";
21
22 // <yes> <report> ONES_HTTP_USAGE http01 <yes> <report>
ONES_BACKDOOR_NETWORK_ACTIVITY netw02

23 Соединение = Новый HTTPСоединение("www.mysite.ru");
```

НДВ: скрытая функциональность (1С)

AVA_VAN.3.3E

Описание

Приложение выполняет код, полученный из строки после декодирования (например, base64). Авторы бэкдоров применяют такую технику, чтобы затруднить обнаружение кода, реализующего недокументированную функциональность.
С точки зрения безопасности, даже если недокументированная функциональность не является намеренно вредоносной, ее наличие дает злоумышленнику дополнительную возможность для успешной атаки на приложение. Например, скрытая функциональность может быть использована для атак, изменяющих поток управления программы.

Пример

В следующем примере приложение выполняет код, обфусцированный с помощью base64-кодирования:
Выполнить(Base64Значение("c3VkbyBybSAtcmYgLyo="));

Рекомендации

- Выясните причины, по которым в коде использовано кодирование. Если это не оправдано с точки зрения функционирования приложения, удалите этот участок кода.

Ссылки

Русскоязычные ссылки

1. Википедия: Программная закладка

Англоязычные ссылки

1. The Art of the Backdoor
2. CWE-912: Hidden Functionality
3. CWE-506

Вхождения

1c/ONES_BACKDOOR_HIDDEN_FUNCTIONALITY.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_BACKDOOR_HIDDEN_FUNCTIONALITY func01
2 Выполнить(Base64Значение("z/Do4uXyIQ=="));
3 // <yes> <report> ONES_BACKDOOR_HIDDEN_FUNCTIONALITY func01
4 Execute(Base64Value("z/Do4uXyIQ=="));
```

1c/ONES_BACKDOOR_HIDDEN_FUNCTIONALITY.bsl:4

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_BACKDOOR_HIDDEN_FUNCTIONALITY func01
2 Выполнить(Base64Значение("z/Do4uXyIQ=="));
3 // <yes> <report> ONES_BACKDOOR_HIDDEN_FUNCTIONALITY func01
4 Execute(Base64Value("z/Do4uXyIQ=="));
```

НДВ: специальная учётная запись (1С)

AVA_VAN.3.3E

Описание

Приложение сравнивает значение переменной, хранящей данные аутентификации, с константным значением. Эта специальная учётная запись может являться частью бэкдора.
Разработчик приложения мог использовать специальную учётную запись (возможно, с повышенными привилегиями) при отладке и оставил соответствующие участки кода в финальной версии, сохранив за собой доступ к функциональности приложения.
Злоумышленник может декомпилировать приложение, извлечь константные строки, задающие параметры специальной учётной записи, и получить доступ к приложению.
Константные параметры (логины, пароли, ключи) не должны храниться в исходном коде приложения.

Пример

В следующем примере приложение предоставляет привилегии администратора пользователю с определёнными логином и паролем:

```
Если ИмяПользователя = "slyDeveloper" И Пароль = "veryLongPassword" Тогда
    ДатьДоступАдминистратора();
КонецЕсли
```

Рекомендации

- Удалите константные логины, пароли, ключи и прочее из исходного кода приложения.
- Храните данные учётных записей в зашифрованном виде в отдельном файле или в базе данных.

Ссылки

Русскоязычные ссылки

1. Википедия: Программная закладка

Англоязычные ссылки

1. CWE-798: Use of Hard-coded Credentials
2. Hardcoded and Embedded Credentials — beyondtrust.com
3. CWE-506: Embedded Malicious Code

Вхождения

1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_BACKDOOR_SPECIAL_ACCOUNT acc001 <yes> <report>
ONES_PASSWORD_HARDCODED pash10

2 Если ИмяПользователя = "Администратор" И Пароль = "пароль" Тогда

3     ДатьНеограниченныйДоступ();
4 КонецЕсли
5 // <yes> <report> ONES_BACKDOOR_SPECIAL_ACCOUNT acc002
```

1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_BACKDOOR_SPECIAL_ACCOUNT acc001 <yes> <report>
ONES_PASSWORD_HARDCODED pash10

2 Если ИмяПользователя = "Администратор" И Пароль = "пароль" Тогда

3     ДатьНеограниченныйДоступ();
4 КонецЕсли
5 // <yes> <report> ONES_BACKDOOR_SPECIAL_ACCOUNT acc002
```

1c/ONES_BACKDOOR_SPECIAL_ACCOUNT.bsl:6

Уровень Средний

Статус Не обработано

```
3     ДатьНеограниченныйДоступ();
4 КонецЕсли
5 // <yes> <report> ONES_BACKDOOR_SPECIAL_ACCOUNT acc002
```

```
6 Если hash = "8743b52063cd84097a65d1633f5c74f5" Тогда


7     ДатьНеограниченныйДоступ();
8 КонецЕсли
```

1c/ONES_PASSWORD_HARDCODED.bsl:24

Уровень Средний

Статус Не обработано

```
21 // <yes> <report> ONES_PASSWORD_HARDCODED pash07
22 FTP.Пароль = "secret";
23 // <yes> <report> ONES_PASSWORD_HARDCODED pash10 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001


24 Если Пароль = "secret" Тогда


25 КонецЕсли;
26 // <yes> <report> ONES_PASSWORD_HARDCODED pash09 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001
27 Если МойПароль = "secret" Тогда
```

1c/ONES_PASSWORD_HARDCODED.bsl:27

Уровень Средний

Статус Не обработано

```
24 Если Пароль = "secret" Тогда
25 КонецЕсли;
26 // <yes> <report> ONES_PASSWORD_HARDCODED pash09 <yes> <report>
ONES_BACKDOOR_SPECIAL_ACCOUNT acc001


27 Если МойПароль = "secret" Тогда


28 КонецЕсли;
29 // <yes> <report> ONES_PASSWORD_HARDCODED pash11
30 Функция СоединениеССервером(Хост, Логин = "admin", Пароль = "123456")
Экспорт
```

Небезопасные настройки политик cross-origin resource sharing (1C)

AVA_VAN.3.3E

Описание

Небезопасные настройки CORS могут привести к компрометации данных.
CORS (Cross Origin Resource Policy) — определённый в стандарте HTML5 механизм, позволяющий JavaScript-коду работать с данными из другого домена. Параметры CORS должны быть определены в HTTP-заголовке Access-Control-Allow-Origin. Недостаточно точно заданный параметр CORS может привести к компрометации данных приложения.

Пример

В следующем примере определён заголовок, открывающий доступ к данным приложения для JavaScript-кода из любого домена:

```
Заголовки = Новый Соответствие;
Заголовки.Вставить("Access-Control-Allow-Origin", "*");
```

Безопасный вариант:

```
Заголовки = Новый Соответствие;
Заголовки.Вставить("Access-Control-Allow-Origin", "www.example.org");
```

Рекомендации

- Задавайте множество доменов, с которых открыт доступ к данным приложения, как можно более точно.

Ссылки

Русскоязычные ссылки

1. Атаки HTML5: что нужно знать — zevvssibirix / habrahabr.ru

Англоязычные ссылки

1. OWASP Top 10 2017-A5-Broken Access Control
2. OWASP: HTML5 Security Cheat Sheet
3. Cross-Origin Resource Sharing — w3.org
4. CWE CATEGORY: OWASP Top Ten 2017 Category A5 - Broken Access Control
5. CWE-346
6. CWE-942

Вхождения

1c/ONES_HTML5_CORS.bsl:3

Уровень Средний

Статус Не обработано

```
1 Заголовки = Новый Соответствие;
2 // <yes> <report> ONES_HTML5_CORS cors01

3 Заголовки.Вставить("Access-Control-Allow-Origin", "*");

4 // <no> <report>
5 Заголовки.Вставить("Access-Control-Allow-Origin", "something");
```

Пустой ключ шифрования (1С)

AVA_VAN.3.3E

Описание

Пустой ключ может привести к компрометации приложения.
Устранить угрозы безопасности, связанные с заданными в исходном коде пустыми ключами, очень сложно. Информация о том, что определённые данные шифруются с пустым ключом, как минимум доступна каждому разработчику приложения. Более того, после того как приложение установлено, удалить из его кода пустой ключ можно только посредством обновления. Константные строки легко извлекаются из скомпилированного приложения декомпиляторами. Поэтому злоумышленнику не обязательно иметь доступ к исходному коду, чтобы узнать о шифровании с пустым ключом.
Уязвимости типа «утечка конфиденциальных данных» (Sensitive Data Exposure) занимают третье место в рейтинге уязвимостей web-приложений «OWASP Top 10 2017».

Пример

В следующем примере пустое значение ключа задано в исходном коде:

```
...
КлючШифрования = "";
...
```

Рекомендации

- Не используйте пустые ключи шифрования.
- Для создания ключей рекомендуется использовать криптографически стойкий ГПСЧ (генератор псевдослучайных чисел).
- Для хранения ключей следует использовать специальные решения, такие как Аппаратные Модули Безопасности (Hardware Security Module).
- В целях безопасности необходимо периодически обновлять ключи.

Ссылки

Русскоязычные ссылки

1. Управление криптографическими ключами
2. Секретные ключи никогда не должны записываться в явном виде на носителе, который может быть считан или скопирован.

Англоязычные ссылки

1. Use of hard-coded password
2. CWE-259: Use of Hard-coded Password
3. OWASP Top 10 2013-A5-Security Misconfiguration
4. OWASP Top 10 2013-A6-Sensitive Data Exposure
5. OWASP Top 10 2017-A3-Sensitive Data Exposure
6. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration

Вхождения

1c/ONES_CRYPTO_KEY_EMPTY.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_KEY_EMPTY keye04

2 publicKey = "";

3 // <yes> <report> ONES_CRYPTO_KEY_EMPTY keye02
4 Процедура СоединениеССервером(Хост, Логин = "admin", ПриватныйКлюч = "")
Экспорт
5     // do smth
```

1c/ONES_CRYPTO_KEY_EMPTY.bsl:4

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_KEY_EMPTY keye04
2 publicKey = "";
3 // <yes> <report> ONES_CRYPTO_KEY_EMPTY keye02

4 Процедура СоединениеССервером(Хост, Логин = "admin", ПриватныйКлюч = "")
Экспорт

5     // do smth
6 КонецПроцедуры
```

Слабое зерно генератора псевдослучайных чисел (1C)

AVA_VAN.3.3E

Описание

Метод, порождающий случайные числа, вызван с целочисленным аргументом, заданным в исходном коде. Соответствующий генератор псевдослучайных чисел (ГПСЧ) порождает предсказуемую последовательность.
Работа многих криптографических алгоритмов основана на использовании

криптографически стойкого ГПСЧ. Некоторые алгоритмы принимают в качестве дополнительного аргумента значение seed и для каждого значения этого параметра порождают предсказуемую последовательность. В таком случае безопасность системы основана на предположении о том, что значения seed будут непредсказуемы. Уязвимости типа «утечка конфиденциальных данных» (Sensitive Data Exposure) занимают третье место в рейтинге уязвимостей web-приложений «OWASP Top 10 2017».

Пример

В следующем примере создаётся предсказуемая последовательность псевдослучайных чисел:
```
ГенераторСлучаныхЧисел = Новый ГенераторСлучайныхЧисел(256);
```

Рекомендации

- Используйте параметр seed полученный из аппаратных источников случайности (тепловой шум (или шум Джонсона), источник радиоактивного распада, генератор свободных колебаний).
- Для работы с ценными данными используйте надежный генератор псевдослучайных чисел, который не порождает предсказуемые последовательности.

Ссылки

Англоязычные ссылки

1. OWASP Top 10 2017-A3-Sensitive Data Exposure
2. OWASP Top 10 2013-A6-Sensitive Data Exposure
3. CWE-1032
4. CWE-331
5. CWE-337

Вхождения

1c/ONES_CRYPTO_BAD_SEED.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_BAD_RANDOM rand01 <yes> <report>
ONES_CRYPTO_BAD_SEED seed01

2 ГенераторСлучаныхЧисел = Новый ГенераторСлучайныхЧисел(256);
```

Слабый алгоритм хеширования (1С)

AVA_VAN.3.3E

Описание

Используемая хеш-функция небезопасна. Её использование может привести к утрате конфиденциальности данных.
Хеш-функции MD2, MD5, SHA1 обладают известными уязвимостями. Нахождение коллизий для функций MD2 и MD5 не требует существенных ресурсов; аналогичная задача была решена для SHA1. Если эти функции применяются для хранения ценной информации (например, паролей), её конфиденциальность может быть нарушена.
Хеш-функция, применяемая для хранения паролей, кроме устойчивости к коллизиям, должна быть не слишком быстрой. Это осложняет атаку путём полного перебора. Для этой цели разработаны специализированные хеш-функции: PBKDF2, bcrypt, scrypt.
Пусть пароли пользователей хранятся на сервере в зашифрованном виде с использованием небезопасной хеш-функции (например, MD5). Возможный сценарий атаки:

1. Злоумышленник получает доступ к базе зашифрованных паролей.
2. Злоумышленник, используя уязвимость алгоритма хеширования, вычисляет строку, для которой алгоритм хеширования даёт то же значение, что и для пароля пользователя.
3. Злоумышленник проходит аутентификацию, используя вычисленную строку.

Уязвимости типа «утечка конфиденциальных данных» (Sensitive Data Exposure) занимают третье место в рейтинге уязвимостей web-приложений «OWASP Top 10 2017».

Пример

Примеры слабого алгоритма хеширования: MD5,SHA1.
ХешMD5 = Новый ХешированиеДанных(ХешФункция.MD5);
ХешSHA1 = Новый ХешированиеДанных(ХешФункция.SHA1);

Рекомендации

- Используйте надёжные функции хеширования (SHA-2).
- Для хеширования паролей используйте специализированные хеш-функции (PBKDF2, bcrypt, scrypt) и полученную из криптографически стойкого генератора псевдослучайных чисел соль.

Ссылки

Русскоязычные ссылки

1. Как надо хешировать пароли и как не надо – d0znpp / habrahabr.ru

Англоязычные ссылки

1. OWASP Top 10 2013-A6-Sensitive Data Exposure
2. OWASP Top 10 2017-A3-Sensitive Data Exposure
3. OWASP: Top 10 2010-A7-Insecure Cryptographic Storage
4. CWE-326: Inadequate Encryption Strength
5. NIST Approved Algorithms
6. How to securely hash passwords – Thomas Pornin / stackoverflow.com
7. MD5 considered harmful today. Creating a rogue CA certificate – Alexander Sotirov, Marc Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik, Benne de Weger / win.tue.nl
8. CWE-328
9. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration

Вхождения

1c/ONES_CRYPTO_BAD_HASH.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_BAD_HASH hash01

2 ХешSHA1 = Новый ХешированиеДанных(ХешФункция.SHA1);

3 ХешSHA1.Добавить(НачальнаяСтрока);
4
5 // <yes> <report> ONES_CRYPTO_BAD_HASH hash01
```

1c/ONES_CRYPTO_BAD_HASH.bsl:6

Уровень Средний

Статус Не обработано

```
3 ХешSHA1.Добавить(НачальнаяСтрока);
4
5 // <yes> <report> ONES_CRYPTO_BAD_HASH hash01

6 ХешMD5 = New DataHashing(HashFunction.MD5);

7 ХешMD5.Добавить(ХешSHA1.ХешСумма);
8
9 Результат = ХешMD5.HashSum
```

1c/ONES_CRYPTO_BAD_HASH.bsl:14

Уровень Средний

Статус Не обработано

```
11 // <yes> <report> ONES_COM_USAGE com002
12 Crypt = Новый COMОбъект("CAPICOM.HashedData");
13 // <yes> <report> ONES_CRYPTO_BAD_HASH hash03
```

```
14 Crypt.Algorithm = 3; // 0 — SHA1; 1 — MD2; 2 — MD4; 3 — MD5; 4 — SHA-256; 5 — SHA-384; 6 — SHA-512.


15
16 // <yes> <report> ONES_CRYPTO_BAD_HASH hash02 <yes> <report> ONES_COM_USAGE com002
17 Crypt = Новый COMОбъект("System.Security.Cryptography.MD5CryptoServiceProvider");
```

1c/ONES_CRYPTO_BAD_HASH.bsl:17

Уровень Средний

Статус Не обработано

```
14 Crypt.Algorithm = 3; // 0 — SHA1; 1 — MD2; 2 — MD4; 3 — MD5; 4 — SHA-256; 5 — SHA-384; 6 — SHA-512.
15
16 // <yes> <report> ONES_CRYPTO_BAD_HASH hash02 <yes> <report> ONES_COM_USAGE com002


17 Crypt = Новый COMОбъект("System.Security.Cryptography.MD5CryptoServiceProvider");


18 // <yes> <report> ONES_CRYPTO_BAD_HASH hash02 <yes> <report> ONES_COM_USAGE com002
19 Crypt = Новый COMОбъект("System.Security.Cryptography.SHA1Managed");
```

1c/ONES_CRYPTO_BAD_HASH.bsl:19

Уровень Средний

Статус Не обработано

```
16 // <yes> <report> ONES_CRYPTO_BAD_HASH hash02 <yes> <report> ONES_COM_USAGE com002
17 Crypt = Новый COMОбъект("System.Security.Cryptography.MD5CryptoServiceProvider");
18 // <yes> <report> ONES_CRYPTO_BAD_HASH hash02 <yes> <report> ONES_COM_USAGE com002


19 Crypt = Новый COMОбъект("System.Security.Cryptography.SHA1Managed");
```

Слабый генератор псевдослучайных чисел (1С)

AVA_VAN.3.3E

Описание

Использованный генератор псевдослучайных чисел (ГПСЧ) небезопасен, так как порождает предсказуемые последовательности. Злоумышленник может обойти систему аутентификации и захватить сессию пользователя, а также осуществить атаку «отравление кэша DNS».
ГПСЧ порождают цепочки чисел на основе начального значения параметра seed. Существует два типа ГПСЧ: статистические и криптографические. Статистические ГПСЧ порождают предсказуемые последовательности, по статистическим характеристикам похожие на случайные. Их нельзя использовать для целей обеспечения безопасности. Результат работы криптографических ГПСЧ, напротив, невозможно предугадать, если значение параметра seed получено из источника с высокой энтропией. Значение текущего времени обладает небольшой энтропией и также небезопасно в качестве seed.
Уязвимости типа «Утечка конфиденциальных данных» (Sensitive Data Exposure) занимают третье место в рейтинге уязвимостей web-приложений «OWASP Top 10 2017».

Пример

В следующем примере приложение порождает предсказуемую последовательность псевдослучайных чисел:
ГенераторСлучаныхЧисел = Новый ГенераторСлучайныхЧисел (ЧислоДляИнциализацииГенератораСлучаныхЧисел);

Рекомендации

- Используйте надежные генераторы псевдослучайных чисел для генерации псевдослучайных чисел в контексте информационной безопасности, например, из .NET.
- Используйте источники с высокой энтропией для генерации параметра seed для ГПСЧ.

Ссылки

Русскоязычные ссылки

1. Подробно о генераторах случайных и псевдослучайных чисел – FallDi / Habrahabr.ru
2. Подводные камни безопасности в криптографии – Bruce Schneier (пер. Василий Кондрашов) / citforum.ru

Англоязычные ссылки

1. OWASP Top 10 2017-A3-Sensitive Data Exposure
2. OWASP: Insecure randomness
3. CWE-330: Use of Insufficiently Random Values
4. CERT: MSC02-J. Generate strong random numbers
5. CWE-497
6. CWE CATEGORY: OWASP Top Ten 2017 Category A6 - Security Misconfiguration
7. CWE-338

Вхождения

1c/ONES_CRYPTO_BAD_RANDOM.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_BAD_RANDOM rand01

2 ГенераторСлучаныхЧисел = Новый ГенераторСлучайныхЧисел
(ЧислоДляИнциализацииГенератораСлучаныхЧисел);
```

1c/ONES_CRYPTO_BAD_SEED.bsl:2

Уровень Средний

Статус Не обработано

```
1 // <yes> <report> ONES_CRYPTO_BAD_RANDOM rand01 <yes> <report>
ONES_CRYPTO_BAD_SEED seed01

2 ГенераторСлучаныхЧисел = Новый ГенераторСлучайныхЧисел(256);
```

Инструкции по настройке WAF

Использование незащищённого протокола HTTP

Описание

Использование HTTP вместо HTTPS позволяет реализовать атаку «человек посередине». Это может привести к полной утрате конфиденциальности передаваемых данных.
Использование протокола HTTPS, основанного на HTTP и SSL / TLS, позволяет защитить передаваемые данные от несанкционированного доступа и изменения. Рекомендуется использовать HTTPS для всех случаев передачи ценной информации между клиентом и сервером, в частности, для страницы авторизации и всех страниц, требующих аутентификации.

Вхождения

1. 1c/ONES_HTTP_USAGE.bsl:2
2. 1c/ONES_HTTP_USAGE.bsl:17
3. 1c/ONES_HTTP_USAGE.bsl:20
4. 1c/ONES_HTTP_USAGE.bsl:23
5. 1c/ONES_PASSWORD_EMPTY.bsl:35
6. 1c/ONES_PASSWORD_HARDCODED.bsl:35

Настройки запуска сканирования

1/1 16.02.2021 21:47:15

Выбрать файлы для **/*

Языки

- [x] ABAP
- [x] Apex
- [x] C#
- [x] C/C++
- [x] COBOL
- [x] Config files
- [ ] Dart
- [x] Delphi
- [x] Go
- [x] Groovy
- [x] HTML5
- [x] Java, Scala, Kotlin
- [x] JavaScript
- [x] LotusScript
- [x] Objective-C
- [x] Pascal
- [x] PHP
- [x] PL/SQL
- [x] Python
- [x] Perl
- [x] Ruby
- [x] Rust
- [x] Solidity
- [x] Swift
- [x] T-SQL
- [x] TypeScript
- [x] VB.NET
- [x] VBA
- [x] VBScript
- [x] Visual Basic 6
- [x] Vyper
- [x] 1C

Настройки Java/Scala/Kotlin

- [x] Не собирать проект (проект уже собран)

Настройки C/C++

- [ ] Проект Visual Studio

Настройки JavaScript

- [ ] Анализировать стандартные библиотеки

Общие настройки анализа

- [ ] Анализировать библиотеки и вложенные архивы
- [ ] Использовать дополнительные правила
- [ ] Инкрементальный анализ

Кодировка исходного кода UTF-8

Кодировка названий файлов UTF-8

Наборы правил —

Настройки экспорта

Информация о проекте

- [x] Динамика уровня безопасности
- [x] Динамика количества уязвимостей

История сканирований

- [ ] Не выгружать историю сканирований
- [x] Выгружать всю историю сканирований
- [ ] Выгрузить последних сканирований ...

Классификация уязвимостей

ОУД4

Информация о сканировании

- [x] Диаграмма найденных уязвимостей
- [x] Диаграмма типов уязвимостей
- [x] Статистика по языкам
- [ ] Статистика по проанализированным файлам
- [x] Информация об ошибках сканирования
- [x] Настройки запуска сканирования

Фильтр уязвимостей

Уровень критичности

- [x] Критический
- [x] Средний
- [ ] Низкий
- [ ] Информационный

Типы уязвимостей

- [x] В стандартных библиотеках
- [x] В .class-файлах, которые не удалось декомпилировать
- [x] С созданной задачей в Jira
- [x] Без инструкций по настройке WAF

Языки

- [x] ABAP
- [x] Android
- [x] Apex
- [x] C#
- [x] C/C++
- [x] COBOL
- [x] Config files
- [x] Dart
- [x] Delphi
- [x] Go
- [x] Groovy
- [x] HTML5
- [x] Java
- [x] JavaScript
- [x] Kotlin
- [x] LotusScript
- [x] Objective-C
- [x] Pascal
- [x] PHP
- [x] PL/SQL
- [x] Python
- [x] Perl
- [x] Ruby
- [x] Rust
- [x] Scala
- [x] Solidity
- [x] Swift
- [x] T-SQL
- [x] TypeScript
- [x] VB.NET
- [x] VBA
- [x] VBScript
- [x] Visual Basic 6
- [x] Vyper
- [x] 1C

Список уязвимостей

Уязвимости со статусами

- [x] Не обработано
- [x] Подтверждено
- [ ] Отклонено

Список вхождений уязвимостей

- ( ) Не выгружать список
- (•) Выгрузить все вхождения
- ( ) Выгрузить вхождений не более ...

Подробные результаты

Уязвимости со статусами

- [x] Не обработано
- [x] Подтверждено
- [ ] Отклонено

Список вхождений уязвимостей

- ( ) Не выгружать список
- (•) Выгрузить все вхождения
- ( ) Выгрузить вхождений не более ...

Источник кода

- ( ) Не выгружать исходный код
- ( ) Выгрузить весь исходный код файла с уязвимостью
- (•) Выгрузить контекст в количестве строк кода 3

Трасса

- ( ) Не выгружать элементы трассы
- (•) Выгрузить только первый и последний элементы
- ( ) Выгрузить все элементы

Дополнительная информация

- [x] Комментарий к уязвимости
- [x] Информация о задачах в Jira

Инструкции по настройке WAF

Инструкции для уязвимостей со статусами

- [x] Не обработано
- [x] Подтверждено
- [ ] Отклонено

Инструкции для WAF

- [x] Imperva SecureSphere
- [x] ModSecurity
- [x] F5

Общие настройки отчёта

- [x] Настройки экспорта отчёта
- [x] Содержание