***Web Services Security***

***UsernameToken Profile 1.0***

***OASIS Standard 200401***

|  |  |
| --- | --- |
| **Термин** | **Определение** |
| SHA | Secure Hash Algorithm |
| SOAP | Simple Object Access Protocol |
| URI | Uniform Resource Identifier |
| UCS | Universal Character Set |
| UTF8 | UCS Transformation Format, 8-bit form |
| XML | Extensible Markup Language |

**1 Introduction**

В этом документе описывается, как использовать UsernameToken со спецификацией WSS: безопасность сообщений SOAP [WSS]. В частности, в нем описывается, как потребитель веб-службы может предоставить UsernameToken в качестве средства идентификации запрашивающей стороны по «имени пользователя» и, при необходимости, с использованием пароля (или общего секрета, или эквивалента пароля) для аутентификации этой личности для производителя веб-службы.

Этот раздел не является нормативным.

**2 Notations and Terminology**

В этом разделе определяются обозначения, пространства имен и терминология, используемые в данной спецификации.

**2.1 Notational Conventions**

Ключевые слова:

"MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL"

«ДОЛЖЕН», «НЕ ДОЛЖЕН», «ТРЕБУЕТСЯ», «ДОЛЖЕН», «НЕ ДОЛЖЕН», «СЛЕДУЕТ», «НЕ ДОЛЖЕН», «РЕКОМЕНДУЕТСЯ», «МОЖЕТ» и «ДОПОЛНИТЕЛЬНО»

В этом документе интерпретироваться, как описано в [RFC 2119]. При описании абстрактных моделей данных в этой спецификации используется соглашение об обозначениях, используемое XML Infoset. В частности, имена абстрактных свойств всегда заключаются в квадратные скобки (например, [какое-то свойство]).

При описании конкретных схем XML [XML-Schema] в этой спецификации используется соглашение об обозначениях WSS: безопасность сообщений SOAP. В частности, каждый элемент свойства [children] или [attributes] элемента описывается с использованием XPath-подобной нотации [XPath] (например, /x:MyHeader/x:SomeProperty/@value1). Использование {any} указывает на наличие подстановочного знака элемента (<xs:any/>). Использование @{any} указывает на наличие подстановочного знака атрибута (<xs:anyAttribute/>). Часто используемые термины безопасности определены в глоссарии Internet Security [SECGLO]. Предполагается, что читатели знакомы с терминами в этом глоссарии, а также с определениями в спецификации безопасности веб-служб.

**2.2 Namespaces**

URI пространства имен (в общей форме «some-URI») представляют некоторые зависящие от приложения или контекста URI, как определено в RFC 2396 [URI]. Эта спецификация разработана для работы с общей структурой сообщений SOAP [SOAP11, SOAP12] и моделью обработки сообщений и должна быть применима к любой версии SOAP. Текущий URI пространства имен SOAP 1.1 используется здесь для предоставления подробных примеров, но нет намерения ограничивать применимость этой спецификации одной версией SOAP. Пространства имен, используемые в этом документе, показаны в следующей таблице (обратите внимание, что для краткости в примерах используются префиксы, перечисленные ниже, но не включаются URI — перечисленные ниже предполагаются).

|  |  |
| --- | --- |
| **Prefix** | **Namespace** |
| ~~SOAP Version 1.1~~ | ~~http://schemas.xmlsoap.org/soap/envelope/~~ |
| SOAP Version 1.2 | http://www.w3.org/2003/05/soap-envelope |
| wsse | http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd |
| wsu | http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd |

URL-адреса, предоставленные для пространств имен wsse и wsu, можно использовать для получения файлов схемы.

**3 Расширения Username Token**

**3.1 Usernames and Passwords**

Элемент <wsse:UsernameToken> представлен в документах WSS: SOAP Message Security как способ предоставления имени пользователя. Внутри элемента <wsse:UsernameToken> может быть указан элемент <wsse:Password>.

Пароли типа wsse:PasswordText и wsse:PasswordDigest не ограничиваются фактическими паролями, хотя это распространенный случай. Можно использовать любой эквивалент пароля, например производный пароль или S/KEY (одноразовый пароль). Имея тип wsse:PasswordText, wsse:PasswordDigest просто подразумевает, что информация, содержащаяся в пароле, находится «в открытом виде», а не содержит «дайджест» информации. Например, если сервер не имеет доступа к открытому тексту пароля, но имеет хэш, то хеш считается эквивалентом пароля и может использоваться везде, где в этой спецификации указывается «пароль». Целью этой спецификации не является требование, чтобы все реализации имели доступ к открытым текстовым паролям. Пароли типа wsse:PasswordText и wsse:PasswordDigest определяются как закодированные в Base64 [XML-Schema] хэш-значения SHA-1 пароля в кодировке UTF8 (или эквивалентного). Однако, если этот дайджест-пароль не отправляется по защищенному каналу или токен не зашифрован, дайджест не обеспечивает реальной дополнительной защиты по сравнению с использованием wsse:PasswordText и wsse:PasswordDigest.

В элемент <wsse:UsernameToken> добавлены два необязательных элемента, обеспечивающих меры противодействия атакам воспроизведения: <wsse:Nonce> и <wsu:Created>. Одноразовый номер — это случайное значение, которое отправитель создает для включения в каждый отправляемый им UsernameToken. Хотя использование одноразовых номеров является эффективной мерой противодействия атакам повторного воспроизведения, оно требует, чтобы сервер поддерживал кэш используемых одноразовых номеров, потребляя ресурсы сервера. Объединение одноразового номера с отметкой времени создания имеет то преимущество, что позволяет серверу ограничивать кэш одноразовых номеров периодом времени «свежести», устанавливая верхнюю границу требований к ресурсам. Если один или оба из <wsse:Nonce> и <wsu:Created> присутствуют, они ДОЛЖНЫ быть включены в значение дайджеста следующим образом:

Password\_Digest = Base64 ( SHA-1 ( nonce + created + password ) )

То есть объединить одноразовый номер, отметку времени создания и пароль (или общий секрет или эквивалент пароля), обработать комбинацию с помощью хеш-алгоритма SHA-1, а затем включить кодировку Base64 этого результата в качестве пароля (обобщение). Это помогает скрыть пароль и обеспечивает основу для предотвращения повторных атак. Производителям веб-сервисов для эффективного предотвращения повторных атак РЕКОМЕНДУЮТСЯ три контрмеры:

1. РЕКОМЕНДУЕТСЯ, чтобы производители веб-сервисов отклоняли любой UsernameToken, не используя как одноразовый номер, так и метки времени создания.

2. РЕКОМЕНДУЕТСЯ, чтобы производители веб-сервисов предоставляли ограничение «свежести» метки времени и чтобы любой UsernameToken с «устаревшими» метками времени отклонялся. В качестве рекомендации можно использовать как минимум пять минут для обнаружения и, следовательно, отклонения повторов.

3. РЕКОМЕНДУЕТСЯ, чтобы используемые одноразовые номера кэшировались на период, по крайней мере, такой же, как период ограничения свежести метки времени, указанный выше, и чтобы UsernameToken с одноразовыми номерами, которые уже использовались (и, таким образом, находятся в кэше), были отклонены. Обратите внимание, что одноразовый номер хэшируется с использованием последовательности октетов его декодированного значения, а метка времени хэшируется с использованием последовательности октетов его кодировки UTF8, как указано в содержимом элемента.

Обратите внимание, что wsse:PasswordDigest можно использовать только в том случае, если простой текстовый пароль (или эквивалент пароля) доступен как запрашивающей стороне, так и получателю. Обратите внимание, что секрет помещается в конец ввода, а не в начало. Это связано с тем, что результатом SHA-1 является полное состояние функции в конце обработки входного потока. Если входной поток точно соответствует размеру блока хеш-функции, злоумышленник может расширить входные данные дополнительными блоками и сгенерировать новые/уникальные хеш-значения, зная только хэш-выход для исходного потока. Если секрет находится в конце потока, то злоумышленники не могут произвольно расширить его, поскольку им приходится заканчивать входной поток паролем, которого они не знают. Точно так же, если одноразовый номер/создан был помещен в конец, злоумышленник может изменить одноразовый номер на одноразовый + созданный и добавить в конце новое время создания для создания нового хэша.

Приведенные выше меры противодействия не охватывают случай, когда токен воспроизводится другому получателю.

Существует несколько (ненормативных) возможных подходов к противодействию этой угрозе, которые можно использовать по отдельности или в комбинации. Их использование требует предварительной договоренности (возможно, в виде отдельно опубликованного профиля, который вводит новый тип пароля) между взаимодействующими сторонами для обеспечения совместимости:

* включение имени пользователя в хэш, чтобы предотвратить случаи, когда несколько учетных записей пользователей имеют совпадающие пароли (например, пароли основаны на названии компании).
* включение имени домена в хэш, чтобы предотвратить случаи, когда одно и то же имя пользователя/пароль используется в нескольких системах.
* включение некоторого указания предполагаемого получателя в хэш, чтобы предотвратить случаи, когда принимающие системы не используют совместно кэш одноразовых номеров (например, два отдельных кластера приложений в одном и том же домене безопасности).

Ниже показан синтаксис XML этого элемента:

<wsse:UsernameToken wsu:Id="Example-1">

<wsse:Username> ... </wsse:Username>

<wsse:Password Type="..."> ... </wsse:Password>

<wsse:Nonce EncodingType="..."> ... </wsse:Nonce>

<wsu:Created> ... </wsu:Created>

</wsse:UsernameToken>

Ниже описаны атрибуты и элементы, перечисленные в примере выше:

/wsse:UsernameToken/wsse:Password

Этот необязательный элемент предоставляет информацию о пароле (или эквивалент, например хэш). РЕКОМЕНДУЕТСЯ передавать этот элемент только тогда, когда используется безопасный транспорт (например, HTTP/S) или если сам токен шифруется.

/wsse:UsernameToken/wsse:Password/@Type

Этот необязательный атрибут URI указывает тип предоставляемого пароля. В таблице ниже указаны предварительно определенные типы (обратите внимание, что фрагменты URI относятся к URI для этой спецификации).

|  |  |
| --- | --- |
| **URI** | **Description** |
| #PasswordText (default) | Фактический пароль для имени пользователя,  хэш пароля, производный пароль или S/KEY.  Этот тип следует использовать при хешировании пароля.  эквиваленты, которые не полагаются на одноразовый номер или создание  время используется, или когда алгоритм дайджеста другой  чем SHA1 используется |
| #PasswordDigest | Дайджест пароля (и, возможно, одноразовый номер и/или временная метка создания) для имени пользователя с использованием алгоритма, описанного выше. |

/wsse:UsernameToken/wsse:Password/@{any}

Это механизм расширения, позволяющий добавлять к элементу дополнительные атрибуты на основе схем.

/wsse:UsernameToken/wsse:Nonce

Этот необязательный элемент указывает криптографически случайный одноразовый номер. Каждое сообщение, включающее элемент <wsse:Nonce>, ДОЛЖНО использовать новое значение одноразового номера, чтобы производители веб-сервисов могли обнаруживать атаки воспроизведения.

/wsse:UsernameToken/wsse:Nonce/@EncodingType

Этот необязательный атрибут URI указывает тип кодирования одноразового номера (допустимые значения см. в определении <wsse:BinarySecurityToken>). Если этот атрибут не указан, используется кодировка Base64 по умолчанию.

/wsse:UsernameToken/wsu:Created

Необязательный элемент <wsu:Created> указывает отметку времени, используемую для указания времени создания. Он определен как часть определения <wsu:Timestamp>.

Все совместимые реализации ДОЛЖНЫ иметь возможность обрабатывать элемент <wsse:UsernameToken>. Если спецификация требует, чтобы элемент был «обработан», это означает, что тип элемента ДОЛЖЕН быть распознан до такой степени, чтобы возвращалась соответствующая ошибка, если элемент не поддерживается.

Обратите внимание, что элементы <wsse:KeyIdentifier> и <ds:KeyName>, как описано в спецификации безопасности сообщений WSS: SOAP, не поддерживаются в этом профиле. Следующий пример иллюстрирует использование этого элемента. В этом примере пароль отправляется в виде открытого текста, поэтому это сообщение должно быть отправлено по конфиденциальному каналу:

<S11:Envelope xmlns:S11="..." xmlns:wsse="...">

<S11:Header>

...

<wsse:Security>

<wsse:UsernameToken>

<wsse:Username>

Zoe

</wsse:Username>

<wsse:Password>

IloveDogs

</wsse:Password>

</wsse:UsernameToken>

</wsse:Security>

...

</S11:Header>

...

</S11:Envelope>

В следующем примере показано использование дайджеста пароля вместе с одноразовым номером и отметкой времени создания:

<S11:Envelope xmlns:S11="..." xmlns:wsse="..." xmlns:wsu= "...">

<S11:Header>

...

<wsse:Security>

<wsse:UsernameToken>

<wsse:Username>

NNK

</wsse:Username>

<wsse:Password Type="...#PasswordDigest">

weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==

</wsse:Password>

<wsse:Nonce>

WScqanjCEAC4mQoBE07sAQ==

</wsse:Nonce>

<wsu:Created>

2003-07-16T01:24:32Z

</wsu:Created>

</wsse:UsernameToken>

</wsse:Security>

...

</S11:Header>

...

</S11:Envelope>

**3.2 Token Reference**

Когда на UsernameToken ссылаются с помощью <wsse:SecurityTokenReference>, атрибут ValueType не требуется. Если указано, ДОЛЖНО быть указано значение <wsse:UsernameToken>.

Следующие форматы кодирования предопределены (обратите внимание, что фрагменты URI относятся к URI для этой спецификации):

|  |  |
| --- | --- |
| **URI Description** | **Description** |
| #UsernameToken | UsernameToken |

Когда на UsernameToken ссылаются из элемента <ds:KeyInfo>, его можно использовать для получения ключа для алгоритма проверки подлинности сообщения с использованием пароля. Этот профиль считает, что конкретные механизмы получения ключей выходят за рамки. Реализации должны согласовать алгоритм получения ключа, чтобы быть совместимыми. Не существует определения KeyIdentifier для UsernameToken. Следовательно, ссылки KeyIdentifier НЕ ДОЛЖНЫ использоваться при ссылке на UsernameToken. Точно так же нет определения KeyName для UsernameToken. Следовательно, ссылки KeyName НЕ ДОЛЖНЫ использоваться при ссылке на UsernameToken. Все ссылки относятся к wsu:Id токена.

**3.3 Error Codes**

Реализации могут использовать пользовательские коды ошибок, определенные в частных пространствах имен, если это необходимо. Но РЕКОМЕНДУЕТСЯ, чтобы они использовали коды обработки ошибок, определенные в спецификации WSS: SOAP Message Security для ошибок подписи, расшифровки, кодирования и заголовка маркера, чтобы улучшить взаимодействие.

При использовании пользовательских кодов ошибок реализации должны быть осторожны, чтобы не создавать уязвимостей безопасности, которые могут помочь злоумышленнику в возвращаемых кодах ошибок.

**4 Security Considerations**

Использование UsernameToken не создает дополнительных угроз, помимо уже идентифицированных для других типов SecurityToken. Атаки повторного воспроизведения можно устранить с помощью временных меток сообщений, одноразовых номеров и кэширования, а также других механизмов отслеживания, специфичных для приложения. Владение токеном подтверждается с помощью ключей, а атаки «человек посередине», как правило, смягчаются. Безопасность на транспортном уровне может использоваться для обеспечения конфиденциальности и целостности как UsernameToken, так и всего тела сообщения. Когда пароль (или эквивалент пароля) в <UsernameToken> используется для проверки подлинности, пароль должен быть надлежащим образом защищен. Если базовый транспорт не обеспечивает достаточную защиту от прослушивания, пароль СЛЕДУЕТ переварить, как описано в этом документе. Тем не менее, пароль должен быть достаточно надежным, чтобы простые атаки с подбором пароля не раскрыли секрет захваченного сообщения.

Когда пароль зашифрован, в дополнение к обычным угрозам для любого шифрования необходимо учитывать две угрозы, связанные с паролем: повтор и угадывание. Если злоумышленник может выдать себя за пользователя, воспроизведя зашифрованный или хешированный пароль, то узнавать настоящий пароль не обязательно. Один из методов предотвращения повторного воспроизведения — использование одноразового номера, как упоминалось ранее. Как правило, также необходимо использовать метку времени, чтобы установить потолок количества предыдущих одноразовых номеров, которые должны быть сохранены. Однако для того, чтобы быть эффективными, одноразовый номер и отметка времени должны быть подписаны. Если подпись также находится над самим паролем до шифрования, то было бы просто использовать подпись для выполнения автономной атаки по подбору пароля. Этой угрозе можно противостоять любым из нескольких способов, в том числе: не включать пароль в подпись (пароль будет проверен позже) или подписывать зашифрованный пароль.

Читателю также следует просмотреть Раздел 13 документа WSS: Безопасность сообщений SOAP для дополнительного обсуждения угроз и возможных контрмер.

Этот раздел не является нормативным.