

Open Specifications Integrate One-Time Passwords with Enterprise Applications

In cooperation with secure systems developers, RSA Security Inc. has started the development of a family of open specifications to simplify the secure integration of One-Time Passwords (OTPs) into applications and infrastructure. These specifications, inaugurating a new OTP series, will enable technology solutions vendors to integrate support for a wide range of OTP methods in an interoperable fashion. They address critical components of OTP technology integration and management, including the initialization of OTP credentials and the retrieval, transport and validation of OTPs.

This technology white paper explains the industry need for open specifications to support OTP integration, describes each of the proposed specifications, discusses the OTP specification process, summarizes RSA Security's support for the specifications and provides a brief glossary of relevant terminology.

TABLE OF CONTENTS

I. THE NEED FOR ONE-TIME PASSWORDS	PAGE 1
II. THE NEED FOR OTP STANDARDS	PAGE 1
III. SUPPORT FOR BOTH DISCONNECTED AND CONNECTED TOKENS	PAGE 2
IV. AREAS OF FOCUS	PAGE 3
Credential Provisioning	PAGE 3
Password Retrieval	PAGE 4
Password Transport and Validation	PAGE 4
V. THE ONE-TIME PASSWORD SPECIFICATION PROCESS	PAGE 6
VI. RSA SECURITY'S SUPPORT FOR THE SPECIFICATIONS	PAGE 6
CONCLUSIONS	PAGE 6
GLOSSARY	PAGE 7

I. THE NEED FOR ONE-TIME PASSWORDS

As organizations migrate more Business-to-Business (B2B) and Business-to-Consumer (B2C) interactions online, the need to protect identities and enable secure remote access has become critical. Traditional “static” passwords are easily stolen, frequently lost and expensive for the enterprise to manage. More complex, “stronger” passwords consisting of arcane and often-lengthy combinations of characters that are changed at regular intervals frustrate users, encourage them to write down their passwords and fuel increased support calls that drive up operational costs.

A One-Time Password (OTP) is a means of more simply and securely proving the identity of a user. In a common implementation model, the end-user carries an authentication device (called a token) that could be a standalone device, such as a card or a fob that can be hung on a key chain. Users could also authenticate via a software solution running in another device, such as a laptop, PDA or phone.

The user’s token and the validation server share a secret (the user’s credential). The token applies an algorithm to this credential to generate the OTP. Key to a successful OTP solution is that the next OTP generated is different from the previous one, and that the sequence of OTPs cannot be guessed based on observation of previous occurrences. The OTPs should only be able to be used once. OTP solutions prove identities more securely because the password for the end-user is always changing.

OTP technology can support strong, two-factor authentication, in which users enter something they know—a Personal Identification Number (PIN)—and something they have—the constantly changing code from a software or hardware token. Organizations are increasingly deploying OTP strong authentication solutions to ensure that only authorized users are granted access to enterprise resources.

OTPs allow organizations to authenticate—prove the identity—of users before letting them login to the network. Today, OTPs are used largely for enterprise access applications, such as logging into the corporate network from a desktop, or via remote access from home or on the road. OTPs meet the security requirements of the enterprise while simplifying the user experience. The use of OTPs for consumer authentication is becoming increasingly common as e-businesses seek to prove the identities of consumers to prevent fraud.

II. THE NEED FOR OTP STANDARDS

There are a variety of OTP methods to consider, including: time synchronous, event synchronous and challenge-response. In each, an algorithm is applied to the credential to generate an OTP. In time synchronous, the algorithm combines time and the credential to generate a new OTP at each time interval (usually a minute). In event synchronous, an OTP is generated in response to an event (usually an end-user pressing a button). In challenge-response, a challenge is entered or sent to the token; this challenge is combined with the credential through an algorithm to create a response (the OTP). Multiple algorithms exist that can accomplish each of these methods, either individually or in hybrid combinations.

Today, a variety of OTP vendors offer a variety of OTP methods. While all methods result in OTPs being generated, the different methods have particular strengths and weaknesses that make some methods better suited for particular enterprise or consumer authentication requirements.

The existing variety of OTP methods as well as the possibility for innovation in both the method type and the supporting algorithms suggest against standardizing on a constrained set of OTP methods and algorithms.

Instead, at a time when demand for OTP solutions is increasing for both enterprise and consumer applications, the industry needs to respond by developing open specifications that allow vendors and enterprises to easily and confidently integrate support for OTPs into their applications while being assured that they will be able to support the variety of OTP solutions that exist today and will continue to exist in the future.

This is why RSA Security is proposing a set of open specifications to simplify the secure integration of OTPs into applications and infrastructures. The initial five specifications in the set are available for public review and comment and can be found at www.rsasecurity.com/rsalabs/otps. The sixth specification will be available soon.

The OTP specifications proposed by RSA Security provide maximum implementation flexibility and they do not constrain implementations to particular OTP algorithms. They will enable technology solutions vendors to more effectively integrate support for OTP technology and are intended to support a wide range of OTP approaches, including time synchronous, event synchronous and challenge-response methods.

These proposed specifications will encourage maximum innovation in OTP methods and algorithms. They address critical components of OTP technology integration and management, including the provisioning, retrieval, transport and validation of OTPs.

III. SUPPORT FOR BOTH DISCONNECTED AND CONNECTED TOKENS

These specifications offer the potential to enable new business models and innovative ways to protect access to information. Historically, OTP solutions have involved end-user devices (tokens), which are not connected to the network or to a client.

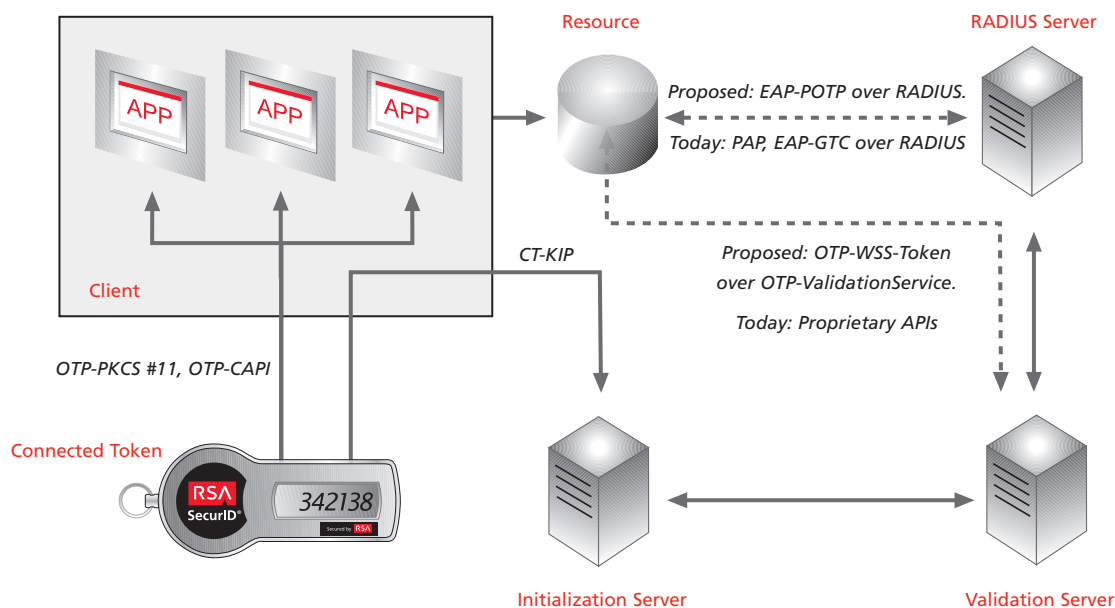
For example, the token could be a hardware device small enough to attach to a key chain. The end-user reads the

OTP from a display and then enters the current value into a computer before the code expires. This approach can be referred to as a “disconnected” approach, because the token is not connected to the network.

While the disconnected approach delivers a high degree of portability, organizations are becoming increasingly interested in supporting “connected” OTP tokens, which deliver increased ease of use and flexibility by enabling a user to authenticate simply by connecting the token. An example of a connected token would be a small USB device that a user can plug into a port on a PC to provide the code without manual data entry. Several of the new OTP specifications are focused on support for connected tokens, while others are relevant to both connected and disconnected tokens.

TEST TYPE	SPECIFICATION	ACRONYM	SUITED FOR	PURPOSE
Credential Provisioning	Cryptographic Token-Key Initialization Protocol	CT-KIP	Connected cryptographic tokens	Allows a token and a server to create and use the same shared credential without passing the credential over the wire
Password Retrieval	PKCS #11 Mechanisms for One-Time Password Tokens	OTP-PKCS#11	Connected OTP tokens	Retrieval and/or validation of OTP from a connected token using the PKCS #11 cryptographic token interface specification
	CryptoAPI Profile for One-Time Password Tokens	OTP-CAPI	Connected OTP tokens	Retrieval and/or validation of OTP from a connected token using the Microsoft CryptoAPI cryptographic token interface specification
Transport and Validation	One-Time Password-Web Service Specification Token	OTP-WSS-Token	All OTP tokens	Formats the OTP information for transport via a web service
	One-Time Password-Validation Service	OTP-Validation Service	All OTP tokens	Will validates OTPs within a web services protocol environment
	Extensible Authentication Protocol-Protected One-Time Password	EAP-POTP	All RSA SecurID® tokens. It also provides a framework for other OTP algorithms to use	Secure transport of OTP information for protocols using EAP, such as PPP, 802.X, IKEv2, etc. It can be used together with or in place of PEAP, EAP-TTLS, FAST etc.

AN OVERVIEW OF HOW THE SPECIFICATIONS COULD BE DEPLOYED IN A STRONG, TWO-FACTOR AUTHENTICATION IMPLEMENTATION



IV. AREAS OF FOCUS

The new OTP specifications fall into the following three areas:

Credential Provisioning

OTP solutions require that both the end-user’s token and the enterprise network’s back-end server share the same credential, which is used to generate the OTP. The specifications in this area are focused on increasing the security and simplicity of the credential provisioning process.

Password Retrieval

OTP retrieval specifications are focused on making it straightforward for vendors to support connected one-time password tokens so end-users can avoid having to manually enter OTPs into applications.

Password Transport and Validation

When integrating OTPs with enterprise applications and infrastructure, it is essential that end-users can enter the OTP for authentication, and that the application or infrastructure can pass that OTP across the network to a validation server.

Credential Provisioning

OTP solutions require that the token and the back-end server share the same credential, which is used to generate the OTP. Today this is often accomplished by the OTP token vendor by seeding a disconnected token with its credential during the manufacturing process and then providing the organization deploying the token with the credential linked in digital format to the serial number of the physical token.

CT-KIP

The Cryptographic Token Key Initialization Protocol (CT-KIP) specification will simplify credential provisioning, enabling organizations to save time and money while also increasing security. This protocol enables the token and the server to create and use the same shared credential—without sending it to each other—and without requiring private key capabilities in a token or an established public key infrastructure.

CT-KIP provides great opportunities for the flexible distribution of tokens. It has potentially far-reaching implications for consumer applications. Tokens can be distributed through retail or online channels, with the enterprise having the capability to initialize them later.

Organizations will be able to provision a shared secret onto a connected token without having to pass that shared secret over the network. Employees, partners or suppliers can be issued uninitialized tokens and the enterprise can later provision the appropriate credentials onto these tokens.

Depending on a token's capabilities, it could be able to store multiple credentials, each provisioned by a different organization when the end-user establishes a relationship with that organization. Users also gain maximum flexibility in getting new tokens if theirs are lost or stolen. CT-KIP provides an open and interoperable means to initialize connected tokens and it enables interoperability between token vendors and provisioning vendors.

Password Retrieval

The OTP retrieval specifications focus on making it straightforward for vendors to support connected one-time password tokens. Today, typical end-users authenticating using common client applications must type their current token code, a PIN and, in certain cases, the next token code as well to gain access.

With these new OTP retrieval specifications, applications can call for OTPs to a connected token using well-known interfaces. The applications can read the code off the connected token, eliminating the need for the end-user to key in this data. These specifications can make it easier for application vendors to support a wide variety of connected tokens. Vendors can even offer additional value-added functions; for example an application can be automatically notified when the connected token is not present. These OTP retrieval specifications are applicable to vendors of applications with clients that need to obtain or authenticate a token code.

With the advent of the following new specifications, application vendors gain maximum flexibility to leverage existing cryptographic interfaces, streamline development and more aggressively incorporate OTP solutions.

OTP-PKCS #11

The Public Key Cryptography Standards (PKCS) are specifications that have been developed by RSA Security in conjunction with systems developers worldwide (such as Microsoft, Apple, Sun, etc.) for the purpose of accelerating deployment of public key cryptography.

PKCS is now ubiquitous in the cryptographic world and is used pervasively in the e-security realm. Applications ranging from web browsers to secure e-mail clients depend on the PKCS specifications to interoperate with one another. PKCS #11 is the cryptographic token interface that defines the technology-independent programming interfaces for cryptographic devices.

PKCS #11 Mechanisms for One-Time Password Tokens (OTP-PKCS #11) allows the standardized retrieval of OTPs from an application using the PKCS #11 interface. PKCS #11 is widely implemented by smart card vendors. The proposed extensions will enable support for OTPs making use of the existing PKCS #11 API calls.

OTP-CAPI

CryptoAPI was developed by Microsoft and is now a widely accepted industry standard for interfacing to cryptographic tokens in the Microsoft environment. It includes signaling and data exchange functionality, and applications such as e-mail clients, web browsers and web servers commonly use CryptoAPI for communications. With the proposed CryptoAPI Profile for One-Time Password Tokens (OTP-CAPI), application vendors will be able to integrate support for a wide variety of OTPs in a standardized fashion, allowing applications to automatically retrieve the OTP credential from the token without requiring users to manually enter the information.

Password Transport and Validation

Traditionally, transport has been accomplished through the use of proprietary or less-than-optimal Extensible Authentication Protocol (EAP) methods such as Generic Token Card (GTC). At the application level, the situation is similar. On the receiving end, the OTP integration has been accomplished through proprietary APIs connecting to validation servers. If a validation server supports the Remote Authentication Dial-In User Service (RADIUS) protocol, transport has been enabled through the encapsulation of EAP within the RADIUS protocol.

These specifications complement the existing RADIUS approach. They enhance the security and flexibility of OTP transport and reduce the need for integration of proprietary APIs by:

- Suggesting a new EAP method suited for modern OTP technology
- Proposing a web service for validation of OTPs

OTP-WSS-TOKEN

Today, vendors offer proprietary APIs for OTP credential validation. These APIs often only support the unstructured transfer of OTP data within proprietary vendor protocols, or else OTPs are transferred as uninterpreted text. As web service-centric vendors look to add support for OTP authentication, the One-Time Password Web Services Specification Token (OTP-WSS-Token) will allow them to represent OTP credentials and associated data in a suitable form for use in web services environments.

The OTP-WSS-Token specification proposes a means to carry OTPs and related information in a web services security token format, which is analogous to the existing web services security username and X.509 certificate token formats.

It will allow end-users to strongly authenticate to a greater number of applications and allow the enterprise to simplify the integration of support for strong authentication by enabling application developers to make use of defined web services interfaces.

OTP-VALIDATION SERVICE

The One-Time Password-Validation Service (OTP-Validation Service) specification will specify how data representing authentication requests can be carried and how results from a validation server can be obtained via a web service interface. This specification will be available in the coming months. The OTP-WSS-Token specification details how OTP information should be packaged up for transport, while the OTP-Validation Service will make use of that format in the context of a validation transaction. It will indicate whether or not the received OTP successfully authenticates its associated user.

EAP-POTP

EAP was originally developed as an extension to the well-established Point-to-Point Protocol (PPP), a popular method of connecting a computer to the Internet using dial-up. EAP is a framework that allows a range of ways for users to authenticate to a relying party. Since its publication, EAP has become common in many network access environments beyond PPP, e.g. for wireless LAN (WLAN) access in accordance with IEE 802.1X or in IPsec credential acquisition (IKEv2).

As a framework, EAP supports multiple authentication methods including tokens, OTPs, digital certificates, public key authentication and smart cards. However, EAP methods focused on OTP technologies are inadequate in today's environment. They were designed for handheld tokens and without support for features crucially needed for added security, such as mutual authentication abilities and establishment of cryptographic session keys.

The EAP-Protected One-Time Password (EAP-POTP) specification describes a general EAP method of supporting OTP tokens and details the use with RSA SecurID® technology and RSA Security's OTP algorithm. It has been designed specifically for supporting OTP requirements and provides several benefits because it:

- Protects OTP values,
- Authenticates the EAP server to the client as well as the client to the server,
- Establishes secret keying material for later use, and
- Is well suited for connected OTP tokens.

EAP-POTP is complementary to-but independent of-EAP tunneling methods such as the Protected Extensible Authentication Protocol (PEAP), Tunneled Transport Layer Security (TTLS) and the Extensible Authentication Protocol-Fast Authentication via Secure Tunneling (EAP-FAST). Organizations that use one of these existing EAP tunneling methods can continue to use them while tunneling EAP-POTP within them to gain the benefits of its support for OTPs.

However, those organizations that have not yet deployed other tunneling methods can rely on EAP-POTP to both create the tunnel and carry the OTP information. The EAP-POTP approach to secure channel creation eliminates the need for a public key infrastructure by relying solely on the OTP for setting up the tunnel, and it provides better protection against "man-in-the-middle" attacks than that provided by existing EAP methods used for OTP-based authentication.

EAP-POTP also offers advantages over other existing authentication methods for network access, such as the Password Authentication Protocol (PAP) or the EAP-GTC method. It offers more robust support for scenarios where a user has forgotten a PIN or for when a user is required to enter an additional OTP to complete their authentication. These scenarios are common in a variety of OTP methods. In addition, EAP-POTP does not pass text strings as prompts for the end user, making it possible for prompt content-when needed-to be based on user preferences.

V. THE ONE-TIME PASSWORD SPECIFICATIONS PROCESS

To further industry collaboration on these proposed specifications, RSA Security is following the same proven process as when the company introduced PKCS in 1991- documents that have since become widely referenced and implemented.

The initial set of six open specifications related to the integration and management of OTPs, (collectively referred to as the One-Time Password Specifications documents) are coordinated online at www.rsasecurity.com/rsalabs/otps and are available for public inspection, review and feedback. The specifications will be further developed through mailing list discussions and occasional workshops, with details available from the OTPS website. In addition, the specifications will be submitted to standards bodies when and as appropriate. The EAP-POTP specification, for example, has already been submitted to the IETF for review.

VI. RSA SECURITY'S SUPPORT FOR THE SPECIFICATIONS

Demonstrating the company's support for these OTP specifications, RSA Security has announced plans to integrate these methods into RSA SecurID® strong authentication technology. Future versions of RSA Security's client for connected RSA SecurID tokens, RSA® Authentication Manager and RSA® Authentication Deployment Manager will support these proposed open specifications. This support will enable RSA Security customers to more easily and cost-effectively integrate and deploy OTP-based strong authentication.

CONCLUSIONS

Perhaps the greatest obstacle to the continued explosive growth of e-business is the confidence with which communicating parties can ensure the identity of the entity at the other end of the network connection. Users need to be confident that they are connected to a legitimate service, and services need to know who is accessing their system. The proposed open OTP specifications detailed in this white paper are intended to stimulate application vendors to integrate support for OTPs into their applications, thereby allowing organizations to leverage strong authentication via OTPs to securely authenticate end-users. These specifications enable the use of both connected and disconnected tokens and they provide maximum flexibility in the provisioning of credentials to the OTP tokens of end-users.

Many organizations have OTP solutions in place today for a portion of their end users interacting with a portion of their applications. As organizations deploy OTP solutions to employees, partners, customers and consumers because of security, compliance and simplicity requirements, they will need an increasing number of applications to support connected and disconnected OTP tokens.

Through development of the proposed open specifications, the OTP specifications intend to facilitate simple, secure integration of OTPs with e-business applications. Please visit www.rsasecurity.com/rsalabs/otps for more information on the status of these proposed specifications. You can download the specifications, sign up for mailing list discussions and find out about workshops.

Learn how technology vendors can leverage these specifications to more effectively integrate support for a wide range of OTP technologies and discover how these specifications successfully address important components of OTP technology integration and management. Please join with RSA Security and other leading secure system developers throughout the industry in developing these open specifications and in standardizing them to enable broad industry support for securing access to enterprise resources using OTPs.

About RSA Security

RSA Security Inc. helps organizations protect private information and manage the identities of people and applications accessing and exchanging that information. RSA Security's portfolio of solutions-including identity and access management, secure mobile and remote access, secure enterprise access, secure transactions and consumer identity protection-are all designed to provide a more seamless e-security experience. Our strong reputation is built on our history of ingenuity, leadership, proven technologies and our more than 15,000 customers around the globe. Together with more than 1,000 technology and integration partners, RSA Security inspires confidence in everyone to experience the power and promise of the Internet. For more information, please visit www.rsasecurity.com.

GLOSSARY

API. Application Programming Interface. A set of published specifications for enabling interoperability, in the context of interfaces rather than protocols.

Authentication. Proving the identity of a user before granting access to enterprise resources.

CryptoAPI. Cryptographic Application Programming Interface. An interface specification developed by Microsoft that allows application developers to add authentication, encoding and encryption to Windows-based applications

Challenge-Response. An OTP method in which a challenge is entered or sent to the token and this challenge is combined with the credential through an algorithm to create a response (the OTP).

Connected Token. An emerging category of OTP tokens which deliver an improved user experience by allowing a user to authenticate to the enterprise without having to manually enter the OTP when the token is electronically connected to the user's IT infrastructure.

Credential. Digital information unique to a particular token that in OTP solutions is acted upon by an algorithm to generate the OTP. The credential is another name for the secret shared between the token and the server.

Disconnected Token. A common form of token that is not electronically connected to the user's IT infrastructure, therefore requiring the user to manually enter the OTP codes to gain access to enterprise information or applications.

EAP. Extensible Authentication Protocol (EAP). An extension to the Point-to-Point Protocol (PPP) that provides a framework for authentication. It supports multiple authentication methods including tokens, OTPs, digital certificates, public key authentication and smart cards.

EAP-FAST. Extensible Authentication Protocol-Fast Authentication via Secure Tunneling. An EAP tunneling method proposed by Cisco Systems.

Event Synchronous. A method in which an OTP is generated in response to an event, such as an end-user pressing a button.

OTP: One-Time Password. A means of more simply and securely proving the identity of a user. The end-user carries an authentication device or token. The user's token and the authentication server share a secret (the user's credential). OTPs support strong, two-factor authentication and they allow organizations to authenticate-prove the identity-of users before allowing them to login to the network.

PEAP. Protected Extensible Authentication Protocol. An EAP tunneling method proposed by Cisco Systems, Microsoft Corporation, et al., with contributions from RSA Security.

PKCS. Public Key Cryptography Standards. Specifications developed by RSA Laboratories in conjunction with secure systems developers worldwide (such as Microsoft, Apple, Sun, etc.) for the purpose of accelerating deployment of public key cryptography. PKCS is now ubiquitous in the cryptographic world and applications ranging from web browsers to secure e-mail clients depend on PKCS to interoperate with one another.

PKCS #11. A token interface standard that defines the technology-independent programming interfaces for cryptographic devices.

PIN. Personal Identification Number. A PIN is used as part of an authentication process.

PPP. Point-to-Point Protocol (PPP). A popular method of connecting a remote computer to the Internet.

Private Key. A secret key in a public key cryptography system that is used to decrypt incoming messages and to sign outgoing messages.

Public Key. The publicly available key in a public key cryptography system that is used to encrypt messages bound for its owner and to verify signatures made by its owner.

RADIUS. Remote Authentication Dial-End-User Service. A widely deployed access, authorization and accounting protocol standard.

Shared Secret. Something that is known by both a token and back-end infrastructure. Also referred to as the end user's credential.

Time Synchronous. An OTP method in which an algorithm combines time and the credential to generate a new OTP each time interval (usually a minute).

Token. A hardware authentication device, or a software authenticator that runs on another platform, such as a laptop, PDA or phone.

Two-Factor Authentication. A strong means of proving identity in which a user enters something he knows (a PIN) and something he-or-she has (the OTP displayed or generated by his-or-her token).

TLS. Tunneled Transport Layer Security. An EAP tunneling method proposed by Funk Software, et. al.



RSA Security Inc.
www.rsasecurity.com

RSA Security Ireland Limited
www.rsasecurity.ie

RSA, RSA Security, SecurID®, the RSA logo and *Confidence Inspired* are either registered trademarks or trademarks of RSA Security Inc. in the United States and/or other countries. All other products and services mentioned are the trademarks of their respective owners.
©2005 RSA Security Inc. All rights reserved.

OTP WP 0205