

Password Management Project Roadmap



This document will guide you through the entire life of a successful password management project, including:

- A needs analysis.
- Who to involve in the project.
- How to select the best product.
- Technical design decisions.
- How to effectively roll out the system.
- How to monitor and assure sound ROI.

Contents

1	Introduction	1
2	Needs analysis	2
2.1	Complexity	2
2.2	User productivity	2
2.3	Support cost	2
2.4	Security violations	3
2.5	OS migration	3
3	Organization	4
3.1	Mandate	4
3.2	Budget	4
3.3	Participants	4
3.4	Ownership	5
4	Selecting a product	6
4.1	Technical requirements	6
4.1.1	Functionality	6
4.1.2	Target systems	7
4.1.3	Integration	7
4.1.4	Deployment	8
4.1.5	Flexibility	8
4.1.6	Security	9
4.2	Vendor profile	10

4.2.1	Financial stability	10
4.2.2	Quality of support	10
4.2.3	Deployment time	10
4.2.4	Single source	10
4.2.5	Future direction	11
4.2.6	Partners	11
5	Project management	12
5.1	Project startup	12
5.2	Product selection	12
5.3	Acquisition	13
5.4	Product deployment	13
6	Post deployment	15
6.1	User adoption	15
6.2	Ongoing support and upgrades	15
6.3	Measuring ROI	15
7	Summary	16

1 Introduction

As today's organizations deploy an ever-growing number of complex systems, password management problems choke help desk systems, cause expensive delays and lost productivity, and threaten to compromise security.

Identifying the cause of these problems, and resolving them, requires the involvement of many interested parties and much strategic planning. Organizations can use a number of software products to address these issues. Selecting the right one also involves taking a number of important factors into consideration.

This document will guide you through the entire life of a successful password management project, including:

- A needs analysis.
- Who to involve in the project.
- How to select the best product.
- Technical design decisions.
- How to effectively roll out the system.
- How to monitor and assure sound ROI.

2 Needs analysis

The first step when selecting and deploying a password management product is to conduct a needs analysis. The needs analysis should identify the problems that a password management system must solve. These should be translated into requirements which the successful vendor must meet.

Following are the most common password management problems, and a brief description of password management functionalities that are required to solve them.

2.1 Complexity

Users frequently have too many passwords on too many different systems. As a result, they either forget their passwords or violate security policy in an effort to remember them.

A password management system should allow users to manage every password from a single screen, and allow users to synchronize their passwords to a single, hard-to-guess password.

2.2 User productivity

Users who forget their passwords waste time on:

- Trying to log in.
- Calling the help desk.
- Waiting for service.
- Proving their identity (authenticating).
- Waiting for a password reset.

Each problem incident may consume 20-30 minutes of user time. In many organizations, users experience this problem 2-4 times annually. In a large user population, this generates a huge volume of user problems and help desk calls.

A password management system should incorporate password synchronization, which helps users to remember their passwords and thus eliminate the majority of password-related problems. It should also include a password self-reset and help desk password reset facility, to speed up the resolution of remaining password problems at the help desk.

2.3 Support cost

Users who forget their passwords call the help desk, and get service. These calls normally represent 20% to 30% of total help desk call volume.

- *Password synchronization can reduce the incidence of password problems.*
- *Self-service password resets help users resolve their own problems, rather than calling the help desk.*
- *A help desk password reset facility should minimize problem resolution time by:*
 - *Integrating caller identification and authentication.*
 - *Supporting password reset on multiple systems from a single screen.*
 - *Automatically creating and closing call records.*

2.4 Security violations

In an effort to remember a large number of passwords, users may violate security policies by:

- Writing down passwords.
- Sharing passwords.
- Selecting easily remembered and guessed passwords.
- Not changing passwords.
- Reusing old passwords.

Password synchronization simplifies and automates the password change process while enforcing security procedures. A password policy engine should ensure that synchronized passwords are strong and changed regularly.

2.5 OS migration

When new network systems are installed, users must be assigned new passwords. When many users are involved, creating new login IDs, assigning each of them an initial password, and securely communicating that password value to the user is a large undertaking.

This process is required in projects such as new OS deployments (for example, migrating to Windows 2000 Active Directory), new authentication services (for example, RADIUS servers supporting many firewalls), and new application deployments (for example, SAP or PeopleSoft deployments).

Password synchronization should allow administrators to assign existing users of new systems a random initial password. Users can then reset some or all of their passwords to a new, known value to gain access to new systems.

3 Organization

To be successful, a password management project must have a mandate, a schedule and a budget. Persons in an organization with a vested interest in password management must be involved early in the project. This ensures that their requirements are met at the design stage, and that they will not object to any part of the project during deployment.

3.1 Mandate

A password management project must start with a clear mandate to solve specific business problems. [Section 2 on Page 2](#) outlines the most likely issues that must be resolved.

Projects that start without this mandate may fail when the time comes to request resources and the support of groups within the organization.

3.2 Budget

It is often helpful to verify, at the onset of a password management project, whether or when sufficient funds will be available. The following items require funding:

- A software license for the selected product.
- Annual support costs.
- Training.
- Hardware and associated software costs (including operating systems, network management software, installation).
- Professional services – to install the selected product and to manage a roll-out.
- Internal resources – for project management, product selection, installation and ongoing system administration and support.

3.3 Participants

Early involvement by all interested parties in an organization ensures that the final design reflects all needs, and that no objections will be raised late in the project.

The following groups are typically involved in a password management project:

- The help desk / I.T. user support: *Must understand how to use the system and its impact on their work. Password management systems typically produce the most tangible cost savings here. Help desk analysts will be the direct users of the system.*

- Desktop support: *Must approve any software that will be installed or executed on workstations, as well as any proposed configuration changes.*
- Systems administrators: *Must understand the impact of a password management solution on the systems they manage.*
- I.T. security: *Must understand the impact on overall security policy and design. Should approve password policies and non-password authentication methods (for example, authentication used for password resets.)*

3.4 Ownership

It is crucial for a password management project to include the system's long-term owner, as early as possible.

Ideally, the long-term system owner and the system's technical administrator(s) will have a strong influence over product selection. These people will have to work with the system and its vendor, so they are more likely to take the time to make a critical analysis of product documentation, and undertake a technical laboratory evaluation of candidate products.

It is risky, on the other hand, to have one team select a product, and a separate team install and manage it.

4 Selecting a product

The ideal password management product should meet all of the project's technical requirements, and be supported by a stable, mature and helpful vendor.

The following sections describe the technical and business requirements that a password management system vendor should meet.

4.1 Technical requirements

4.1.1 Functionality

A password management system should include functionality for:

- **Password synchronization:**

Users should be able to maintain a single password that applies to most, if not all, of their login IDs. This helps users remember their passwords and reduces calls to the help desk. Users with a single password are less likely to write down or share their passwords.

- **Self-service password reset:**

Users who forget their passwords should be able to quickly resolve their problems without calling the help desk.

- **Help desk password reset:**

Support analysts should be able to authenticate a caller, reset passwords and automatically create or close a help desk ticket from a single screen. This reduces call duration and cost, and improves customer service.

- **Multiple access methods:**

Users should be able to access the system using all methods offered by the organization, including:

- A web browser and an existing password change user interface, for routine password changes.
- A web browser from the user's own desktop login screen, for self-service password resets.
- An interactive voice response (IVR) system, for users who need to reset their remote access password.

- **Profile builders:**

- In some organizations, users have different login IDs on different systems. If no database exists to correlate IDs to users prior to deployment, then a profile builder must be available to collect this information from users.
- If users will be authenticated for password resets using personal information profiles, then a profile builder may be required to update existing data (for example, in a human resources database) or to create new authentication profiles.

In most cases, the user profile builders should be tools included in the password management authentication module.

- **Language support:**

Organizations that use languages other than English should be able to deploy the solution with multiple languages.

4.1.2 Target systems

A successful password management system should be able to manage passwords on most or all of the systems to which users login with an ID and password.

If this is not possible, then a threshold for systems that must be supported should be defined. A reasonable approach is to require the solution to manage passwords for systems that generate 95% of password-related calls to the help desk.

Target systems should work “out of the box” in as many cases as possible.

Where this is infeasible (e.g., home-grown applications, vertical market applications, legacy applications), the product should be open enough to make it possible to easily integrate with applications:

- Some applications include an API for managing passwords. While rare, this is a useful mechanism to integrate a password management system. It's useful to check the language bindings of any such API, and compare these to what the password management system supports.
- Some applications include command-line tools to manage passwords. The password management system should be able to execute these – on whatever platform they are available.
- Some applications store their passwords in a database, where a password management system may manipulate them directly. This includes client/server applications and web applications with DBMS back-ends.
- Some applications run on midrange or mainframe systems, and can be manipulated by scripting interaction with a terminal login session.
- Some applications present a web GUI, and a password management system can interact with them by simulating the actions of a web browser.

4.1.3 Integration

A password management system should integrate seamlessly with existing I.T. infrastructure, including:

- **Authentication systems:**

Users should be able to authenticate using existing infrastructure – be it a network login ID/password (such as a Windows NT domain), security tokens (such as an RSA SecurID) or by answering questions drawn from an H.R. database.

- **Support systems:**

The system should automatically create issues / tickets in any help desk's support system used by the organization (such as Remedy or Peregrine).

- **Electronic mail:**

The system should be able to interact with users by e-mail – for example, to prompt them to register, or notify them of events related to their login IDs.

- **Telephony:**

Users should be able to access a self-service password reset using existing telephony servers.

- **System monitoring:**

Existing infrastructure should be able to monitor the password management server health, and react to alarm conditions.

4.1.4 Deployment

Deployment should be as simple as possible. Features supporting this objective include:

- **No use of any desktop software components:**

Even very small and simple desktop software must be deployed to thousands of PCs in a large organization. These PCs may not conform to corporate standards, and an installation process that works for one may fail on another.

Clearly, it is preferable to avoid desktop software deployment entirely, and eliminate the related risks, effort and expenditure.

- **Minimize server agents:**

Installing agents on a production server normally involves a lengthy change control process. Using existing client software to communicate with servers reduces deployment time.

- **Integrate with existing databases:**

A password management system should take advantage of existing user profile databases, which may include information such as a list of which systems each user logs into, or what questions to ask a user to authenticate him if he forgot his password.

- **Automatic discovery of login information:**

The system should automatically detect new or deleted login IDs on the systems where it manages passwords. This reduces both initial deployment and the ongoing administration effort.

- **Self-service registration:**

Users should be able to update their own profiles in the system, including login IDs and authentication data.

4.1.5 Flexibility

The system should cope with both current and possible future requirements for:

- **User interface:**

The user interface should be customizable, and support different appearances for different users (such as multiple languages or user groups).

- **Help desk integration:**
The business logic of updating information in a help desk system should be customizable.
- **Password policy:**
The system must be able to enforce a global password policy.
- **Password reset authentication policy:**
The system must support the organization's policy for authenticating users who require a password reset.

4.1.6 Security

A password management system literally owns the “keys to the kingdom” and consequently must meet the most stringent security requirements:

- **Encryption**
 - User access to the system must be encrypted, across every user interface where this is feasible (a notable UI where this is not feasible is the telephone).
 - Any sensitive data stored in the system should likewise be encrypted or hashed, as appropriate. This includes administrative passwords of people authorized to manage the product, as well as passwords used by the product to manage target systems. This also includes any sensitive user profile data (e.g., authentication Q&A).
 - The product should support encrypted communication with all target systems – including those that do not natively implement an encrypted client/server protocol (e.g., most DBMS servers, mainframes, etc.).
 - Encryption should rely on well-known implementations of well-known, trusted encryption and hashing algorithms.
 - Encryption keys should be managed effectively. For example, public keys must be signed by a real certificate authority (and not by the vendor). Private keys must be obscured and protected by operating system ACLs.
- **Authentication**
 - Users must be properly authenticated for every system access. This is done, for example, by asking users to answer multiple personal questions, by having users type their password to some trusted system, or using hardware tokens.
Some measures that are clearly **not** secure enough include:
 - * PINs – which can be guessed, may be intercepted in e-mail distribution, and are likely to be forgotten by users in any case.
 - * Use of a single challenge/response question.
 - Administrators must be duly authenticated prior to getting access to the system. They should use the most secure means possible – e.g., hardware tokens or strong passwords. Q&A profiles are generally not strong enough to be suitable for use by administrators.
- **Accountability** The system must record every possible event, so that users and administrators alike can be held accountable for their actions.

- **Hardened platform**

- The product should operate on a locked down operating system.
- The product should support a diversity of web servers, so that if a given web server is deemed to have an unacceptable history of vulnerabilities, it can be avoided.
- The product should be accessible across web proxies, so that it can be installed in a protected subnet, and accessed across a firewall without opening non-HTTPS ports.
- The product should not require the installation of (possibly insecure or vulnerable) client software.

4.2 Vendor profile

As with any vendor, the company supporting a password management system should offer sound support, effective professional services, good relationships with other relevant vendors, and long term stability.

4.2.1 Financial stability

The five vendors with the largest market share in password management products are all small, and with one exception privately held corporations.

In the interests of long term support for the technology, it is important to verify that prospective vendors are financially sound: growing rather than shrinking, and profitable rather than burning cash reserves.

4.2.2 Quality of support

Quality technical support is crucial to project success. This is best measured by implementing the password management system in a test environment, and evaluating the ability of the vendor to assist in the installation process.

4.2.3 Deployment time

Vendors should be able to offer turn-key or assisted deployments. A good vendor will be able to successfully deploy the system in a minimum amount of time. A good product can be deployed without intrusion – without installing desktop software, and with limited use of server agents.

The deployment effort in a large organization should not take more than 10-20 supplier person/days.

4.2.4 Single source

It is easier and safer to work with a vendor that can provide all the required technology directly. This eliminates the risks of using third party technology, such as:

- Increased cost.
- Uncertain future product availability and revision.
- Limited, poor or inconsistent technical support.

4.2.5 Future direction

The successful vendor should have a clear direction for future growth and technology advancement. This helps to ensure vendor stability, and a sound future for the product.

4.2.6 Partners

Password management products must inter-operate with other I.T. infrastructure supported by current suppliers. Relationships between a password management vendor and the vendors of other infrastructure or services can streamline interoperability and ongoing support.

In particular, it is helpful if the password management vendor has a working relationship with providers of:

- Support portal technology.
- I.T. and help desk outsourcers.
- Security infrastructure.

5 Project management

The following sections outline the objectives of each phase in a password management deployment project.

5.1 Project startup

To begin the project:

1. Perform a needs analysis, as described in [Section 2 on Page 2](#).
2. Document technical and business requirements, as described in [Section 4 on Page 6](#).
3. Establish a project whose mandate is to resolve the problems identified in the needs analysis.
4. Identify prospective vendors and products.
5. Allocate and approve people, systems and a budget.

5.2 Product selection

To make an effective product selection:

1. Perform some research to find out what products are currently available. Analyst firms generally know which vendors have significant market share, and can identify prospects.

Another excellent source of information is the Internet: use a search engine to find sites that mention:

- “password synchronization,”
- “self-service password reset,”
- “help desk password reset”, and
- “password management.”

2. Once you have identified prospective vendors, forward your technical and business requirements document to them, and request a proposal.
3. Provide the prospective vendors a list of key decision-makers in your organization and their selection criteria. This will help vendors to focus their efforts on what matters most to you.
4. Evaluate the product in either a laboratory environment or with a pilot group of users and systems. Evaluating products based on paper only is very risky. You may reach final conclusions based on unfounded or inaccurate information.

Vendor RFP responses are no substitute for lab testing: some vendors will respond to RFPs based on what they believe the customer wants to hear, with no bearing on what their product can actually do, on the theory that “we can either build it later, or convince the customer that they don’t need it.”

Analyst reports are also no substitute for lab testing: the analysts do not install products in their own labs, and instead rely on every vendor for an assessment of their own capabilities. Specific vendor claims are not verified.

Ensure all features defined in the requirements document are tested and compared. This exercise will highlight differences between products and vendors in a way that a paper process cannot.

5. Compare vendor proposals, technical evaluation results and prices.

5.3 Acquisition

Once a product has been selected, negotiate on a price and project deliverables and sign a contract. Fixing the price and deliverables (professional services, milestones, level of support) mitigates project risk.

Include a detailed list of deliverables and a statement of work attached to the contract.

5.4 Product deployment

Prepare a detailed deployment plan including: system design, schedule and resource allocation. These should cover the following aspects:

1. **Design:**

Determine:

- Which features will be activated.
- How users will access the system.
- Which security policies (such as authentication process, password policy) will be enforced.
- Whether the system will integrate with the help desk issue tracking system, and if so how/when it will create open/closed tickets.
- Whether the system will integrate with e-mail, the events that will trigger e-mail, and the messages to be delivered.
- Whether the system will integrate with an authentication database, and the database and schema to be used.
- Whether the system will include meta directory integration, and the direction, directory, and attributes to be used.
- The number of servers needed.

2. **Installation:**

Determine how you will carry out:

- Operating system and web server installation.
- Application software installation.
- Integration with the help desk system, meta directory, H.R. database, e-mail, authentication systems, etc.
- Multi-server replication.

3. **Pilot test:**

Determine how you will carry out the pilot test by:

- Deploying the system to a limited number of users.
- Verifying the functionality of the system.

- Ensuring that the (possibly customized, possibly multi-lingual) user interface is easy to understand.
- Verifying that all interfaces work as expected.

4. Training:

Determine how you will:

- Train help desk analysts to use the tool, and to assist users with using it.
- Compose training materials for the user population, to be posted on the Intranet and e-mailed directly to users.

5. User roll-out:

Determine how you will notify users about the tool, and (if required) activate them.

Be sure to get supporting documentation and best practices from the vendor.

6 Post deployment

While the bulk of the work in a password management deployment process ends with user roll-out, more work is required on an on-going basis. This includes:

6.1 User adoption

- System logs should be analyzed periodically to measure utilization – how many users access the software, and how often.
- Users who have not used the system should be prompted to do so, ideally using an automated process.
- If there are persistent user adoption problems, users can be encouraged to use the system by offering prizes, or required to use the system as a matter of policy.

6.2 Ongoing support and upgrades

A technical resource must be assigned to ongoing system support. In particular, this person must:

- Monitor the system.
- Act as an advocate for the system, to encourage utilization.
- Answer user and help desk questions.
- Periodically add target systems.
- Troubleshoot any problems that may arise.
- Alter integration business logic as help desk, authentication, meta directory and e-mail systems are changed.
- Install software upgrades.

A mature product should allow to minimize the amount of effort required to perform these duties.

6.3 Measuring ROI

- System logs can be used to determine the incidence of help desk and self-service password resets initially and over time.
- Logs can also be used to calculate the average time required by users and by support analysts to resolve password problems on-line.
- This data should be used to support the initial project cost, in terms of reduced problem frequency, reduced use of support resources, and faster problem resolution.

7 Summary

Password management systems offer a simple way to improve user service, reduce network security vulnerabilities, and lower I.T. support costs.

A typical project can go through concept, needs analysis, technical requirements, product selection, installation, pilot testing and roll-out in six months or less. Positive return on investment is typically achieved within 6 months of general roll-out.