presented by

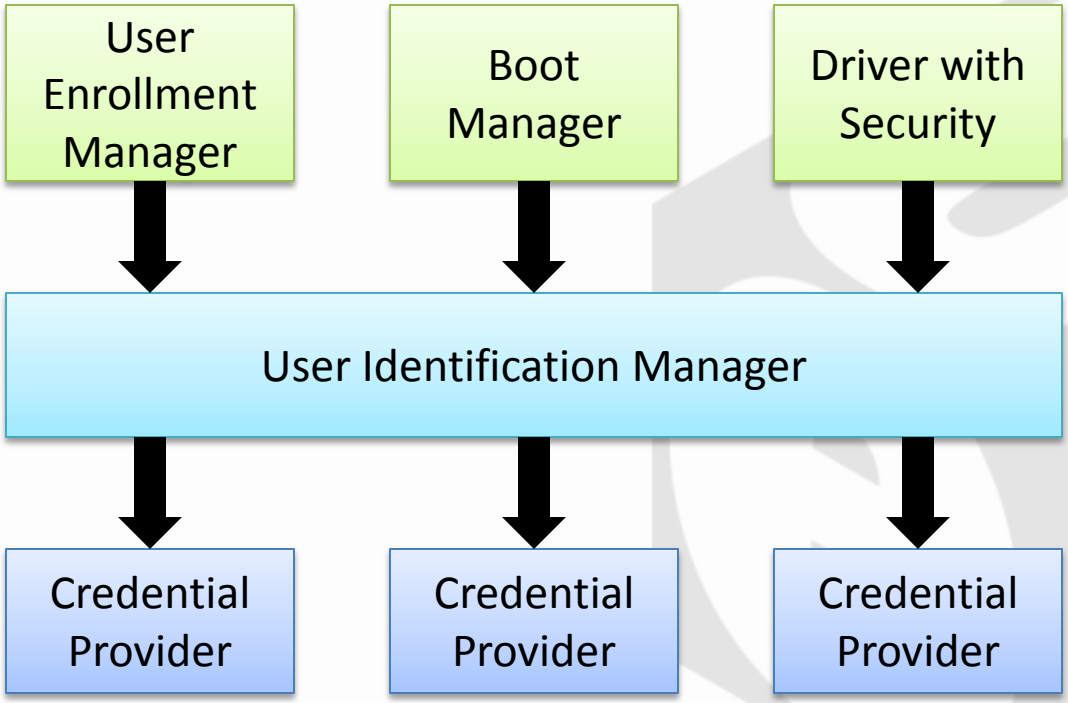
UEFI User Identification

UEFI Winter Plugfest – February 20-23, 2012
Presented by Dick Wilkins (Phoenix)

Agenda

- Architectural Overview
- User Identity Manager
- User Enrollment Manager
- Identification
- Credential Drivers
- Consuming Security Policy

Architectural Overview



User Identity Manager

The User Identity Manager has two roles:

- Provide the User Profile Database
- Drive the Identification Process

The User Manager Protocol is used by:

- User Enrollment Manager
- Boot Manager
- Drivers with Security Policy

User Enrollment Manager

Manages User Profiles, providing a user interface:

- Add, Modify or Delete a User Profile
- Browse all User Profiles
- Browse all Credential Providers
- Associate Credential Providers with User Profiles.

Identification

User Manager response to Indentify ().

The User Manager will either:

- Determine the User Profile to attempt based on prior knowledge.
- Scan all Credential Drivers to see if any of them assert that they provide a default user.
- Ask the user which User Profile to use.

Once a User Profile is selected the User Manager will walk through all of the Credential Drivers required by the profile:

- Call Select () to let the driver prepare.
- Call Form () to get the HII data for user interaction.
- Send the HII data to the Form Browser.
- Call User () to see if the Credential Provider indentified the user.

After all the Credential Providers required by the User Profile have successfully identified the user, the User Manager must then walk all the Credential Providers to add any User Information available at the Credential level to the User Profile.

Credential Drivers

A Credential Driver provides one or more factors for identification of the User. Factors can be:

- something you know
- something you have
- something you are

Examples of Credential Providers:

- Password (know)
- Bluetooth ID (have)
- Smart Card (have)
- Fingerprint (are)
- Face Recognition (are)

Consuming Security Policy

1. Locate the User Manager Protocol (ump).
2. Get the profile of the current user, ump->Current ().
3. Loop through the User Info Handles, ump->GetNextInfo ().
4. Get each handle with ump->GetInfo () to find the handle with InfoType == EFI_USER_INFO_ACCESS_POLICY_RECORD.
5. Loop through the records to answer the policy question.

There are 9 policies defined in the Specification.

EFI_USER_INFO_ACCESS_FORBID_LOAD	0x00000001
EFI_USER_INFO_ACCESS_PERMIT_LOAD	0x00000002
EFI_USER_INFO_ACCESS_ENROLL_SELF	0x00000003
EFI_USER_INFO_ACCESS_ENROLL_OTHERS	0x00000004
EFI_USER_INFO_ACCESS_MANAGE	0x00000005
EFI_USER_INFO_ACCESS_SETUP	0x00000006
EFI_USER_INFO_ACCESS_FORBID_CONNECT	0x00000007
EFI_USER_INFO_ACCESS_PERMIT_CONNECT	0x00000008
EFI_USER_INFO_ACCESS_BOOT_ORDER	0x00000009

Consuming Security Policy

If the EFI_USER_INFO_ACCESS_POLICY_RECORD does not provide a definition of the policy, EFI_USER_INFO_GUID_RECORD allows for extending the policies with GUID defined policies.

The steps to search for a GUID defined policies:

1. Locate the User Manager Protocol (ump).
2. Get the profile of the current user, ump->Current ().
3. Loop through the User Info Handles, ump->GetNextInfo ().
4. Get each handle with ump->GetInfo () to find the handle with InfoType == EFI_USER_INFO_GUID_RECORD.
5. Compare this GUID to the policy GUID.

If all the User Info Handles are processed and the GUID is not found then the policy is not present.

Call to action

- Phoenix believes that the UEFI spec provides a good basis for User Identification and credential management
- There are some questions about implementation that may drive spec updates
- We would like to engage with stakeholders to work on inconsistencies
- Contact me and we can get a dialog going Dick_Wilkins@phoenix.com

Thanks for attending the
UEFI Winter Plugfest 2012

For more information on the Unified EFI Forum and UEFI Specifications, visit
http://www.uefi.org

presented by