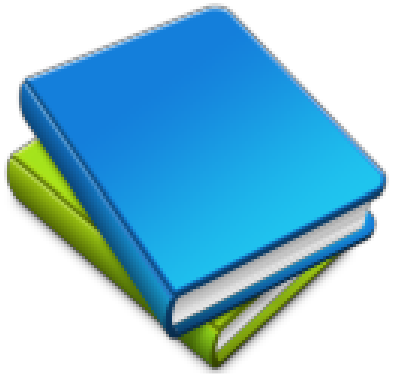
Windows OS Security

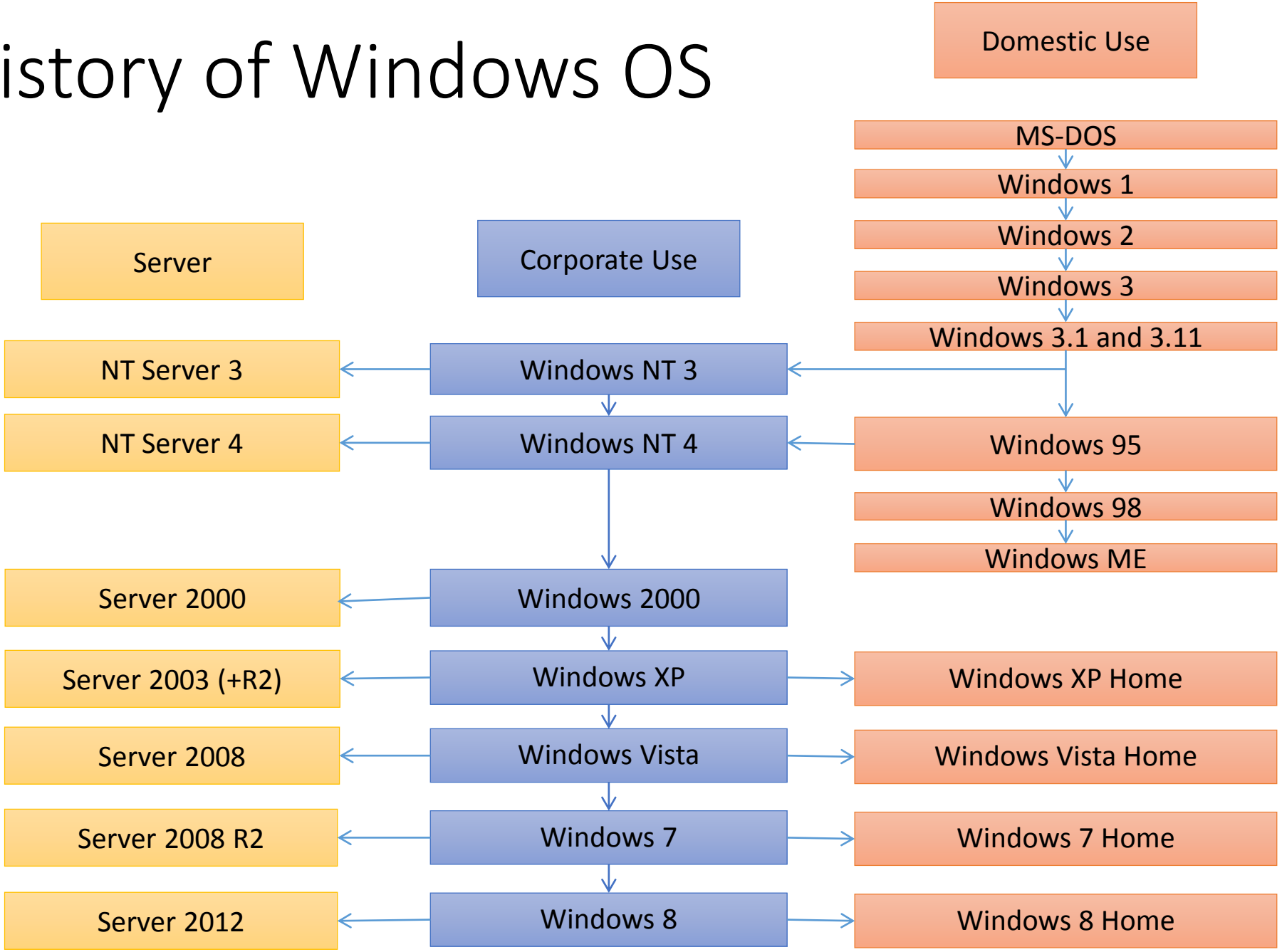
Windows OS and Windows Security Model

Table of Contents

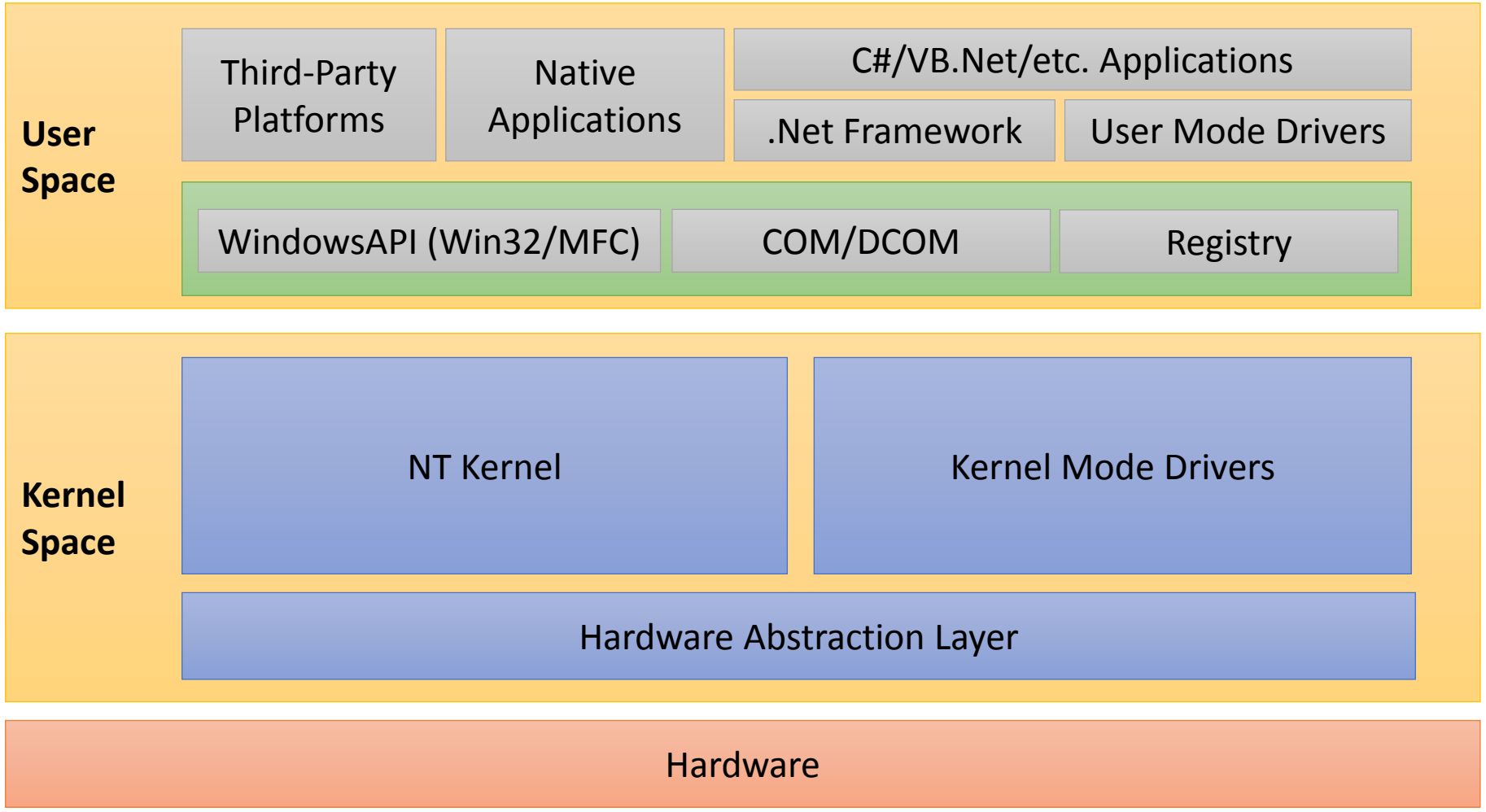


Windows OS

Brief History of Windows OS



Windows OS High Level Architecture



Windows as dev platform: The old way

- Win32
- MFC
- COM
- DCOM

Windows as dev platform: Nowadays

- .Net Framework
- Open Standards
- Open Protocols
- Web Services

Windows as dev platform: Subsystem for UNIX-based Applications (SUA)

- Source-compatibility subsystem for compiling and running custom UNIX-based applications
- Multi-user UNIX environment on Windows
- It offers true UNIX-based functionality without any emulation
- You cannot cross subsystems SUA vs NT/Win32
- Command line only (No X applications)

❖*There is also an open source (GPL) project "Cygwin" that provides unix-like environment for Windows. SUA and Cygwin are based on different architecture.*

Windows Security Model

Accounts and Security Principals

Accounts

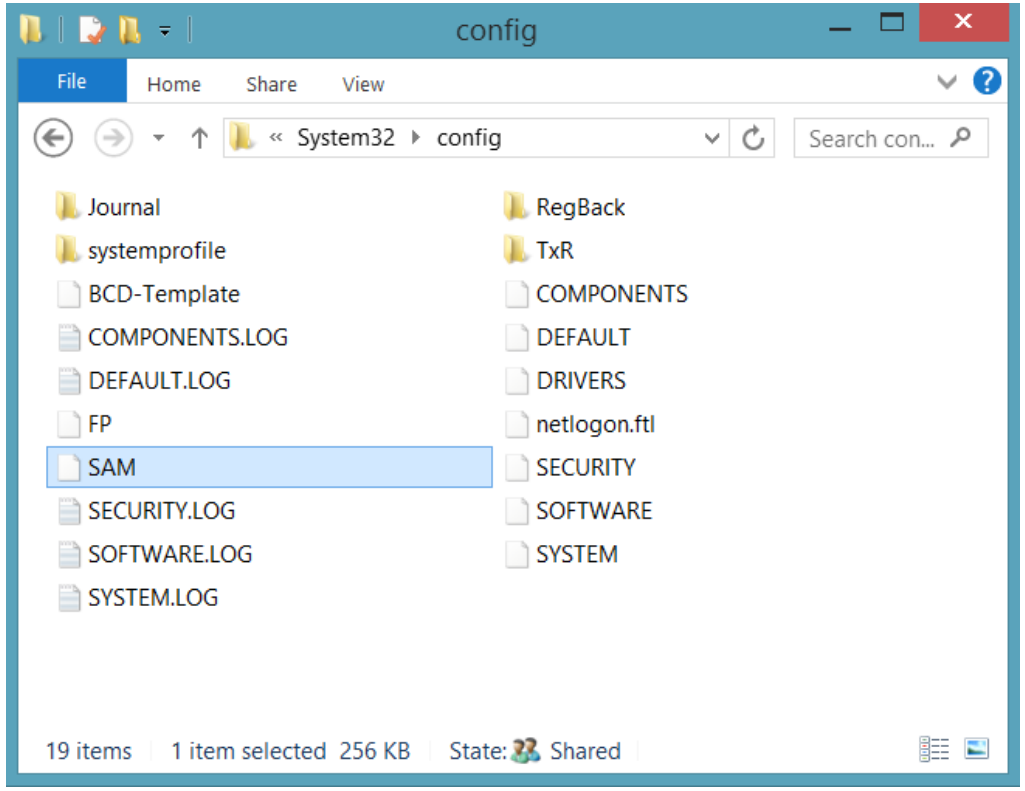
- What does Account mean?
  - What about computer accounts….
- Why we need accounts?
  - Everyday we use various services to do our job or to enjoy.
- How we protect our accounts?
  - Usually we use username and password

Authentication and Authorization

- **Authentication refers to a process that verify who you are.**

- **Authorization refers to a process that verify what you are authorized to do.**

Security Account Manager

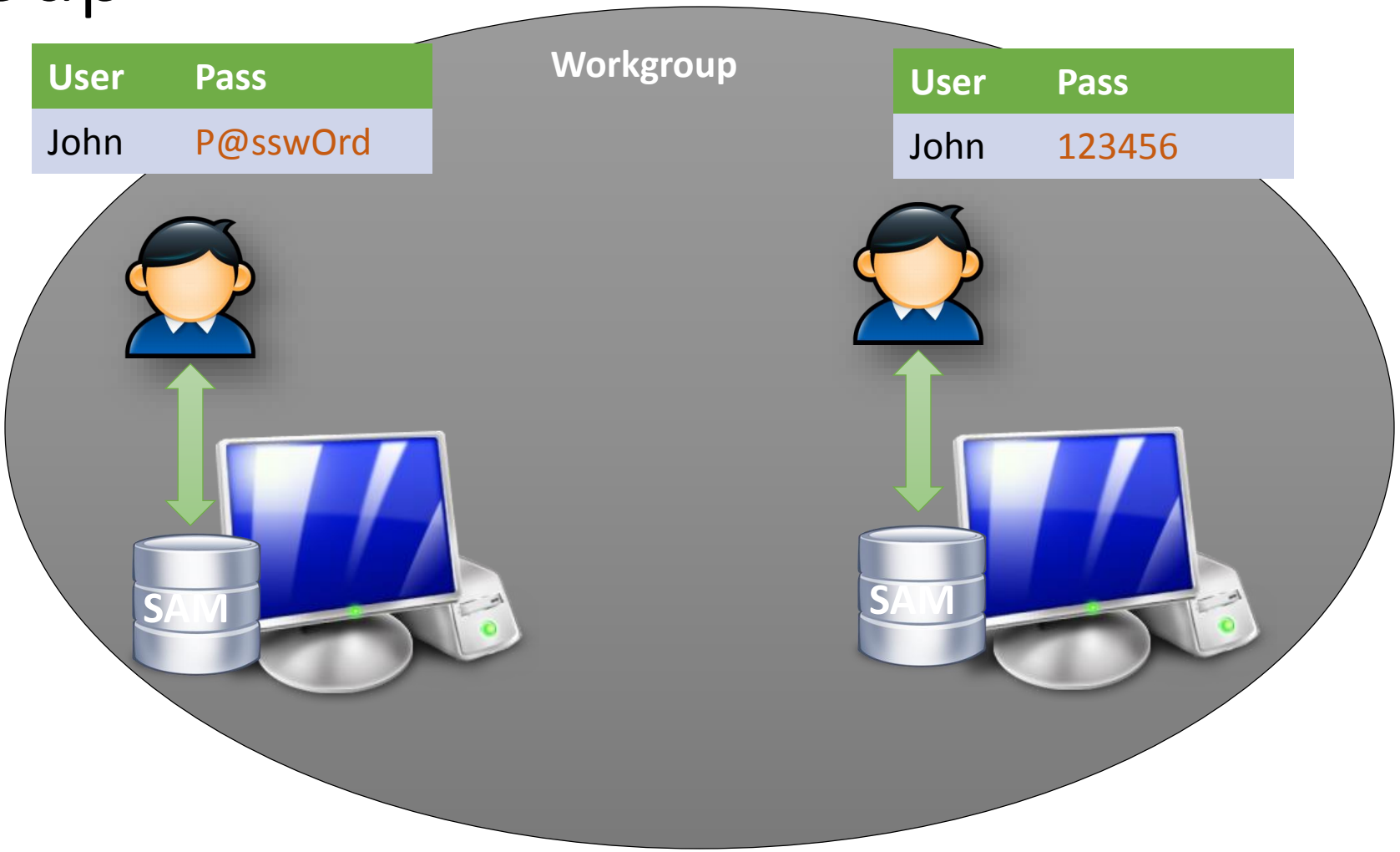


- A registry hive that stores:
  - User accounts
  - Groups
  - Security information
- Accessible only by system processes

Active Directory Domain Service

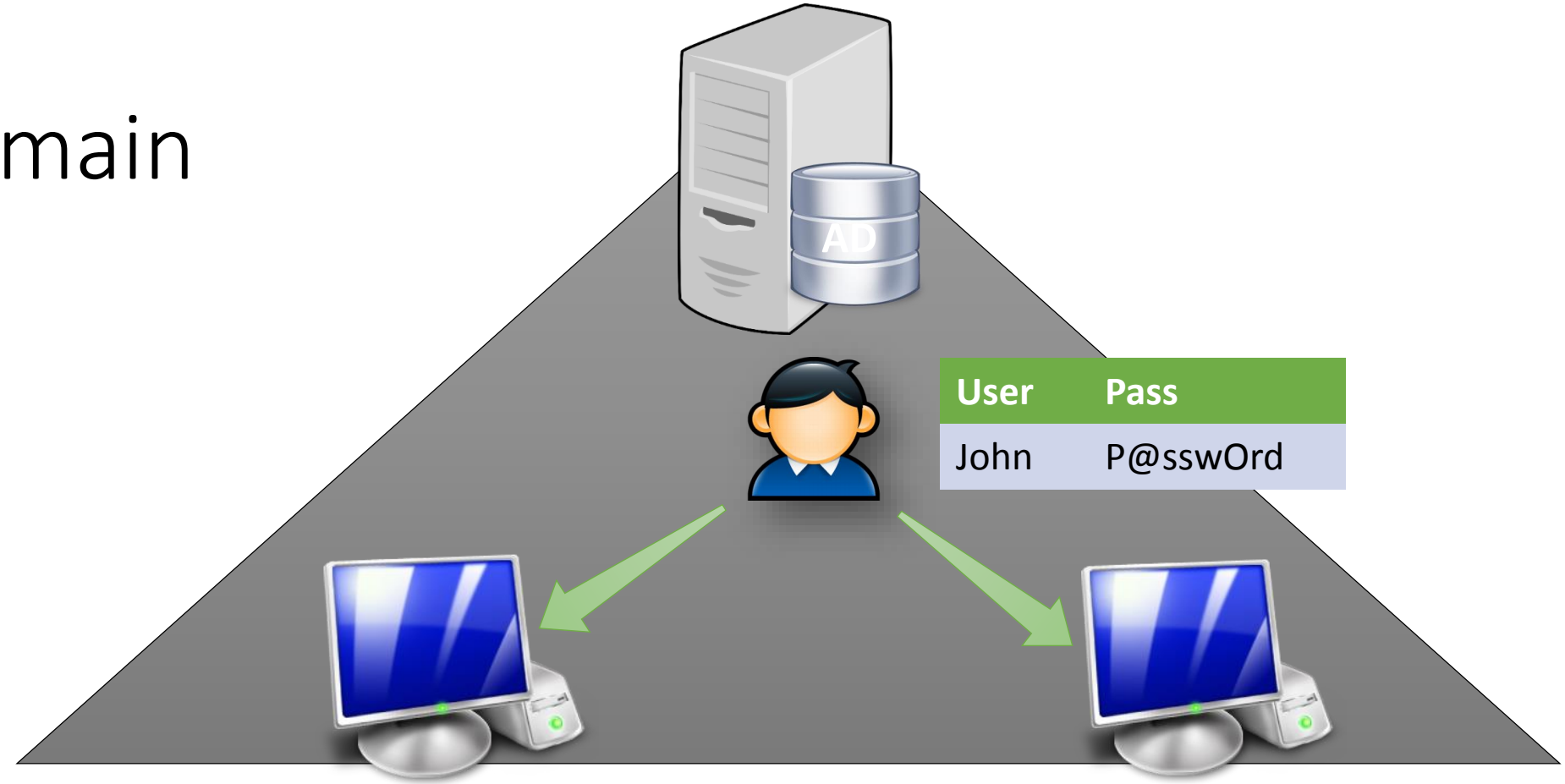
- Stores accounts (User + Computer) information in a central database
- Organizes various objects into a hierarchical tree
- Provides information for network resources
- Enforces security polices
- Audit

Workgroup



- Each computer has local SAM database
- Suitable for small networks 2-10 computers

Domain



- Accounts are stored in a central database
  - More secure
  - More Scalable
  - Easy to manage

Security Principals

- Entities that the windows security system recognizes
- Foundation for controlling access to securable resources
- Domain and Local
  - Domain
    - User Accounts
    - Computer Accounts
    - Groups
    - Well-known security principals
  - Local
    - User Accounts
    - Groups

Security Identifier (SID)

- Windows creates automatically a Security Identifier (SID) for each **security principal**
  - S-1-5-21-AAA-BBB-CCC-RRR
  - Security Identifiers are always unique
  - Windows uses Security Identifier to recognize you
  - You can think for SID as Personal ID Number (EGN)

Access Tokens

What is an Access Token?

An access token contains the security information for a logon session

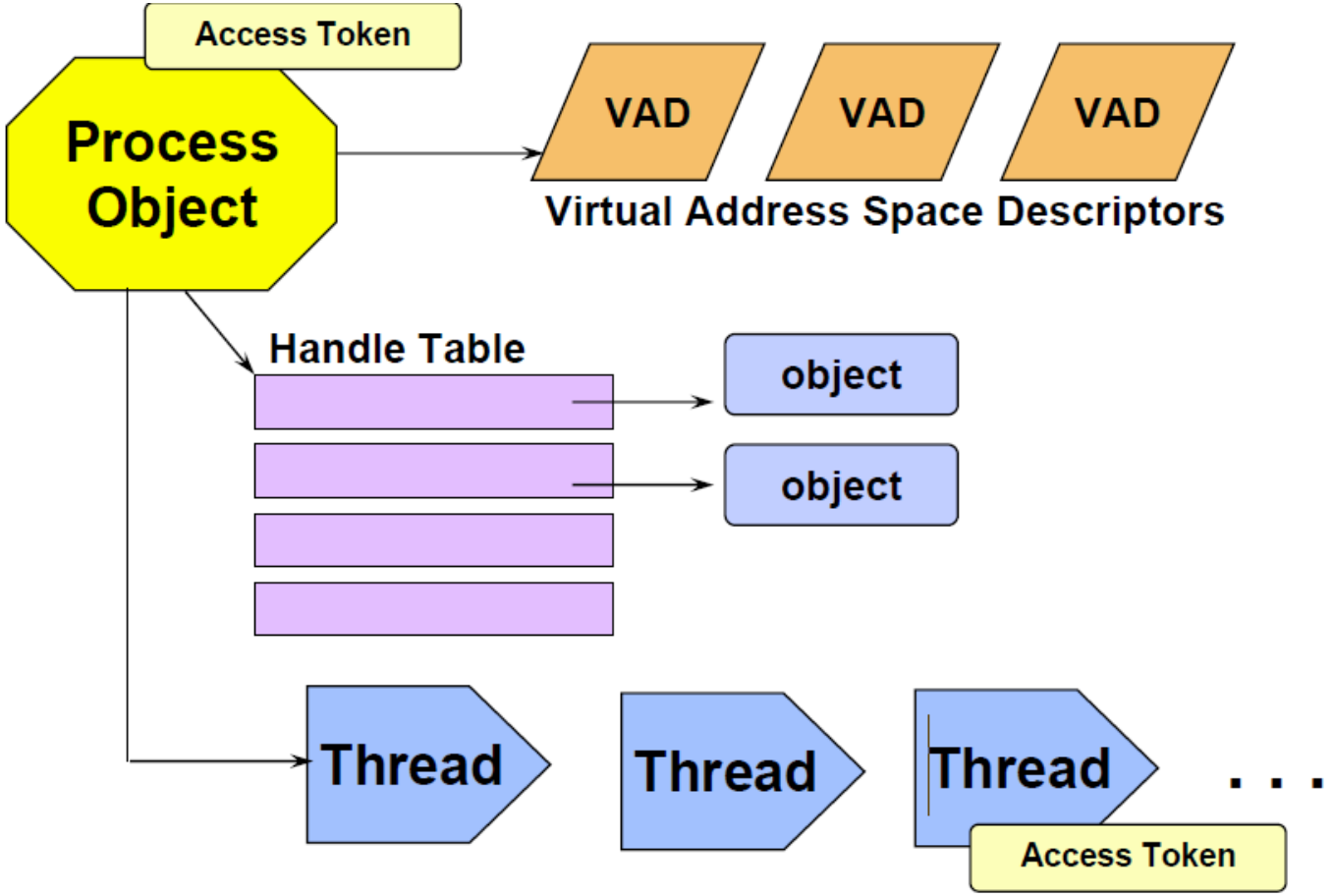
- The system creates an access token when a user logs on
- Every process executed on behalf of the user has a copy of the token
- The system uses the token to control access to securable objects

What information contains an Access Token?

- User SID
- Groups Membership SIDs
- Privileges
  - System-wide permissions assigned to the logon user account

❖*In Windows 2012, Microsoft introduced a new feature Dynamic Access Control which extends the access token with addition information*

Processes, Threads and Access Tokens



Demonstration

- How to validate your access token
- You have to logoff and logon again to update the information in your access token

Security Descriptors (SD) and Access Control Lists (ACL)

Security Descriptors

- Security Descriptors are data structures of security information
  - Who is the owner of this object?
  - Who have access to read/write/etc?
  - Are parent access rules included yes/no?
- Security Descriptors can be associated with different OS objects
  - File System objects
  - Registry objects
  - Windows Service

Access Control Lists (ACL)

- Objects needing protection are assigned an ACL that includes
  - SID of object owner
  - List of access control entries (ACEs)
- Each ACE includes a SID and Access Mask
  - Access mask could include
    - Read, Write, Create, Delete, Modify, etc.

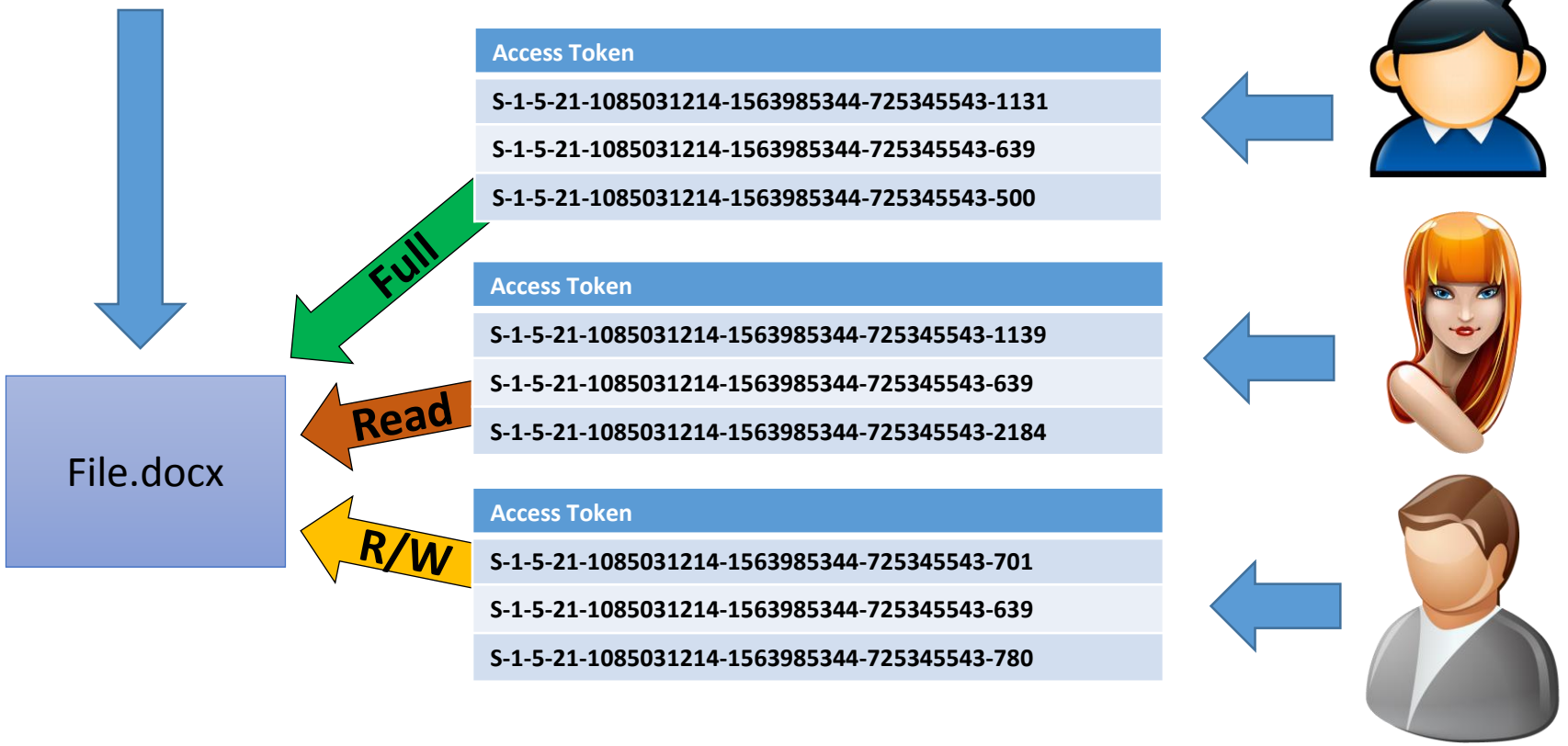
❖ *The Access Mask is different for each type of object (e.g. File, Printer, Registry etc.*

Access Control Lists (ACL) (cont.)

- Discretionary ACL
  - Grants or denies access to protected resources such as files, directories, network shares, shared memory, etc.
- System ACL
  - Used for auditing

Access Control Process

Group/User	Type
S-1-5-21-1085031214-1563985344-725345543-780	R/W
S-1-5-21-1085031214-1563985344-725345543-639	Read
S-1-5-21-1085031214-1563985344-725345543-500	Full



Demonstration

- File System Permissions
- Registry Permissions

❖ *Because of the object nature of Windows, ACLs can be associated with any object created by NT Object subsystem*

Logon Process

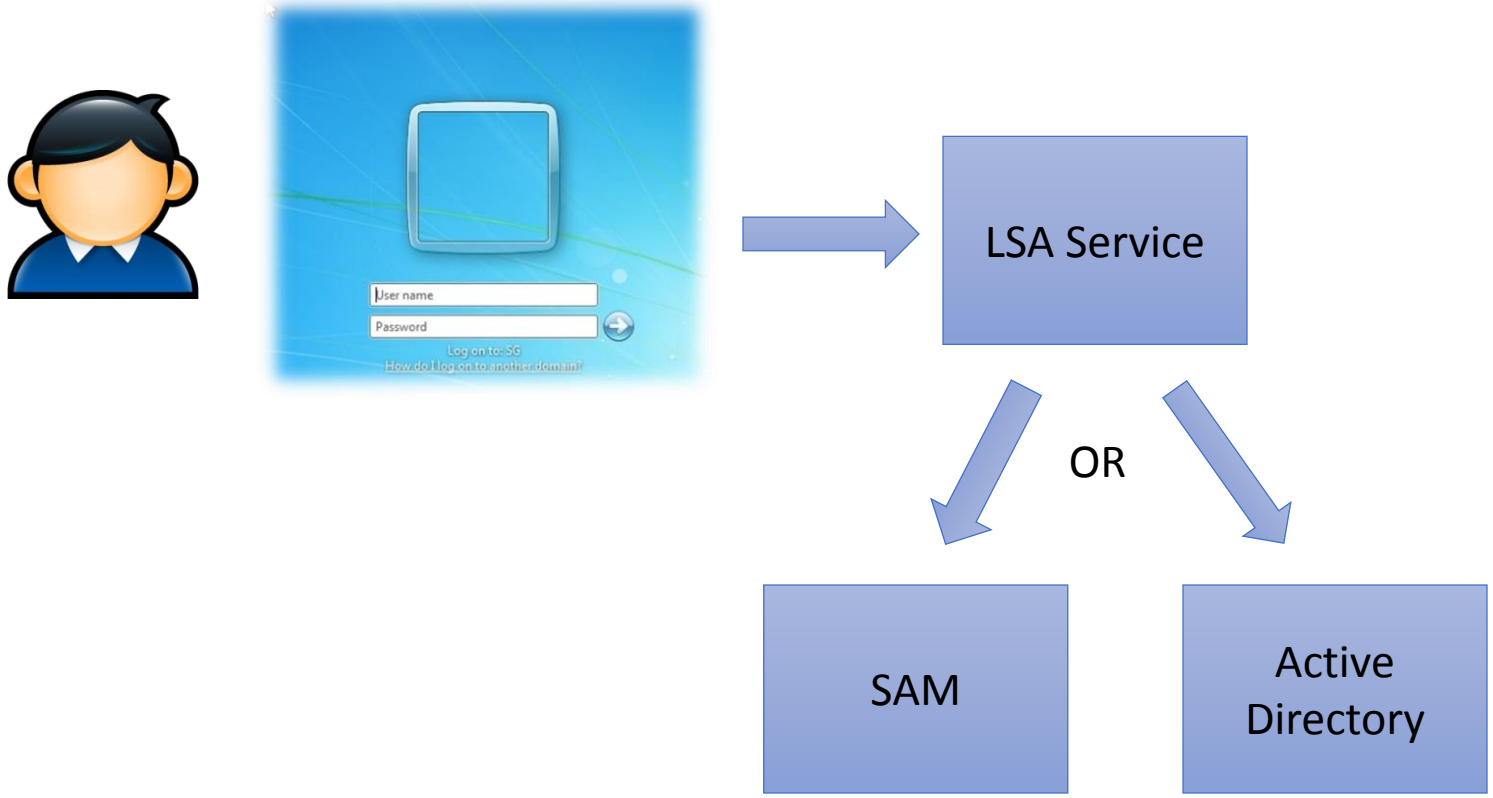
Logon Process

- **Interactive Logon (WinLogon)**

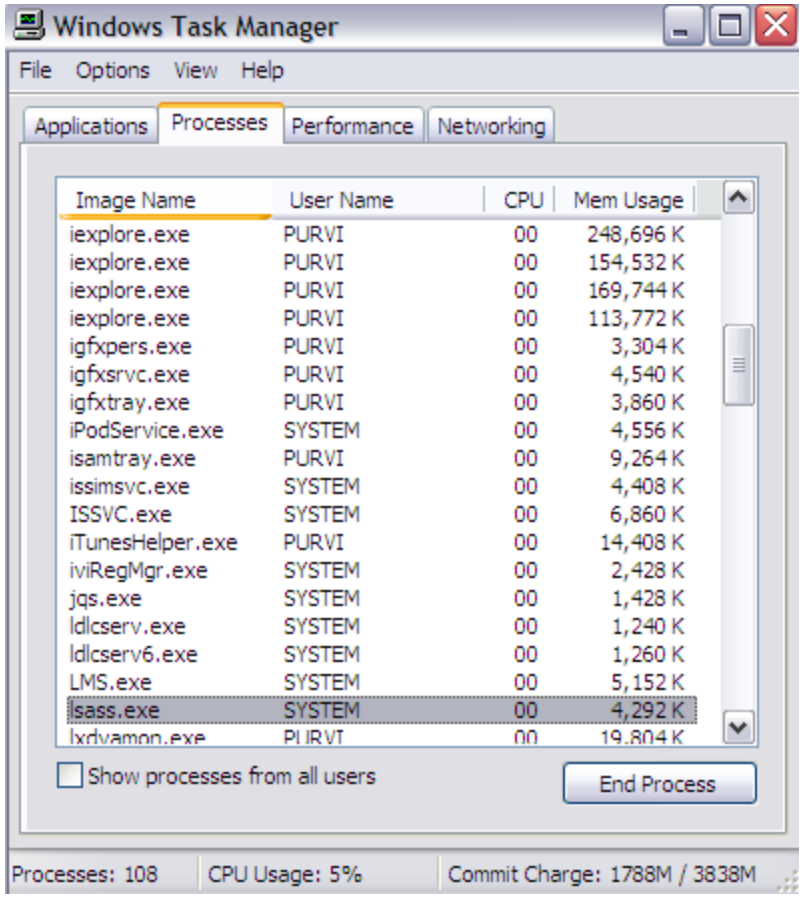
- **Network Logon (NetLogon)**

Log on

- The interactive logon process is the first step in user authentication and authorization.

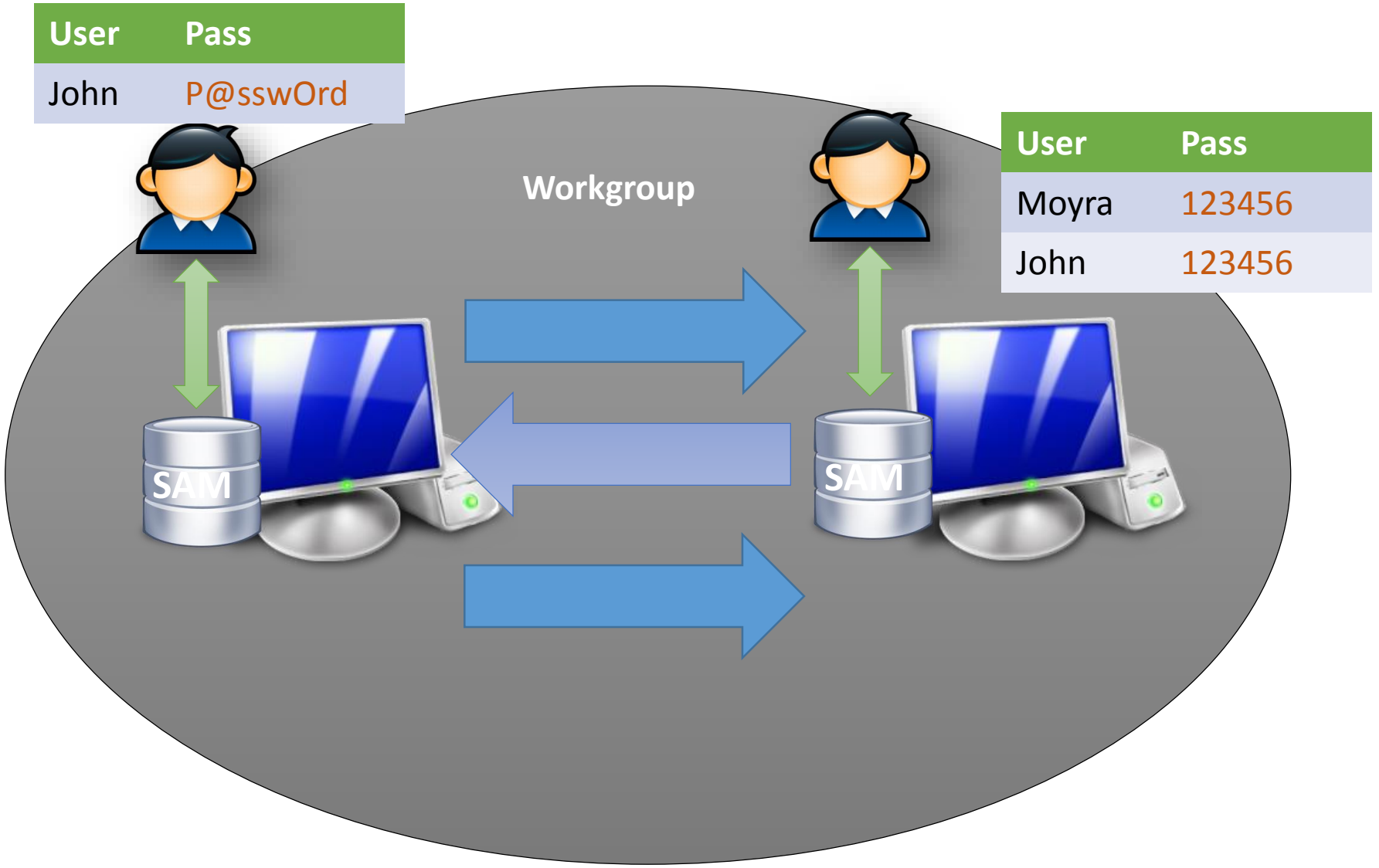


Local Security Authority (LSA)

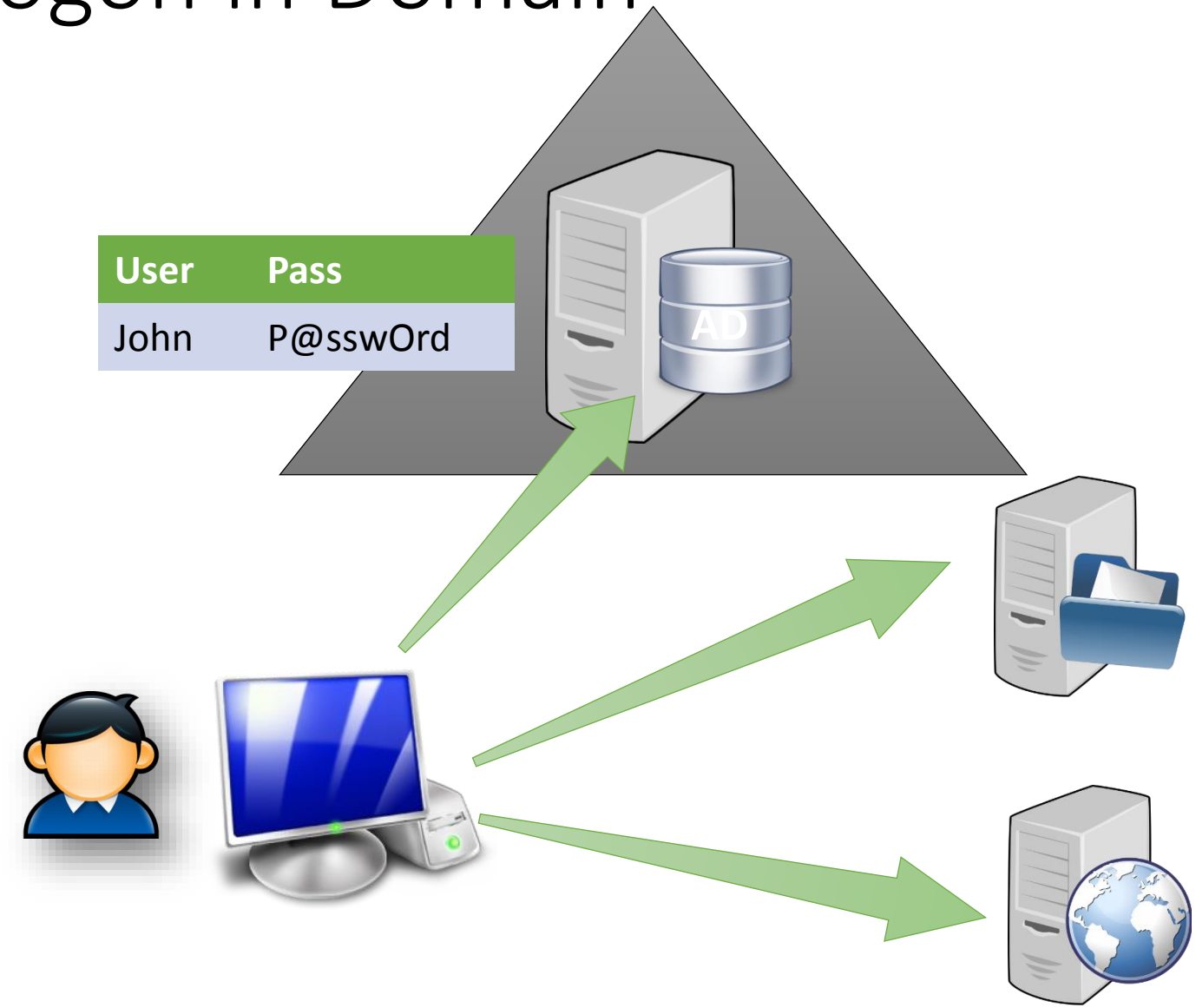


- Issues security access tokens to accounts
- Responsible for enforcing local security policy
  - Lsass.exe
  - User mode
- Key component of the logon process

Network Logon in Workgroup



Network Logon in Domain



Authentication Protocols

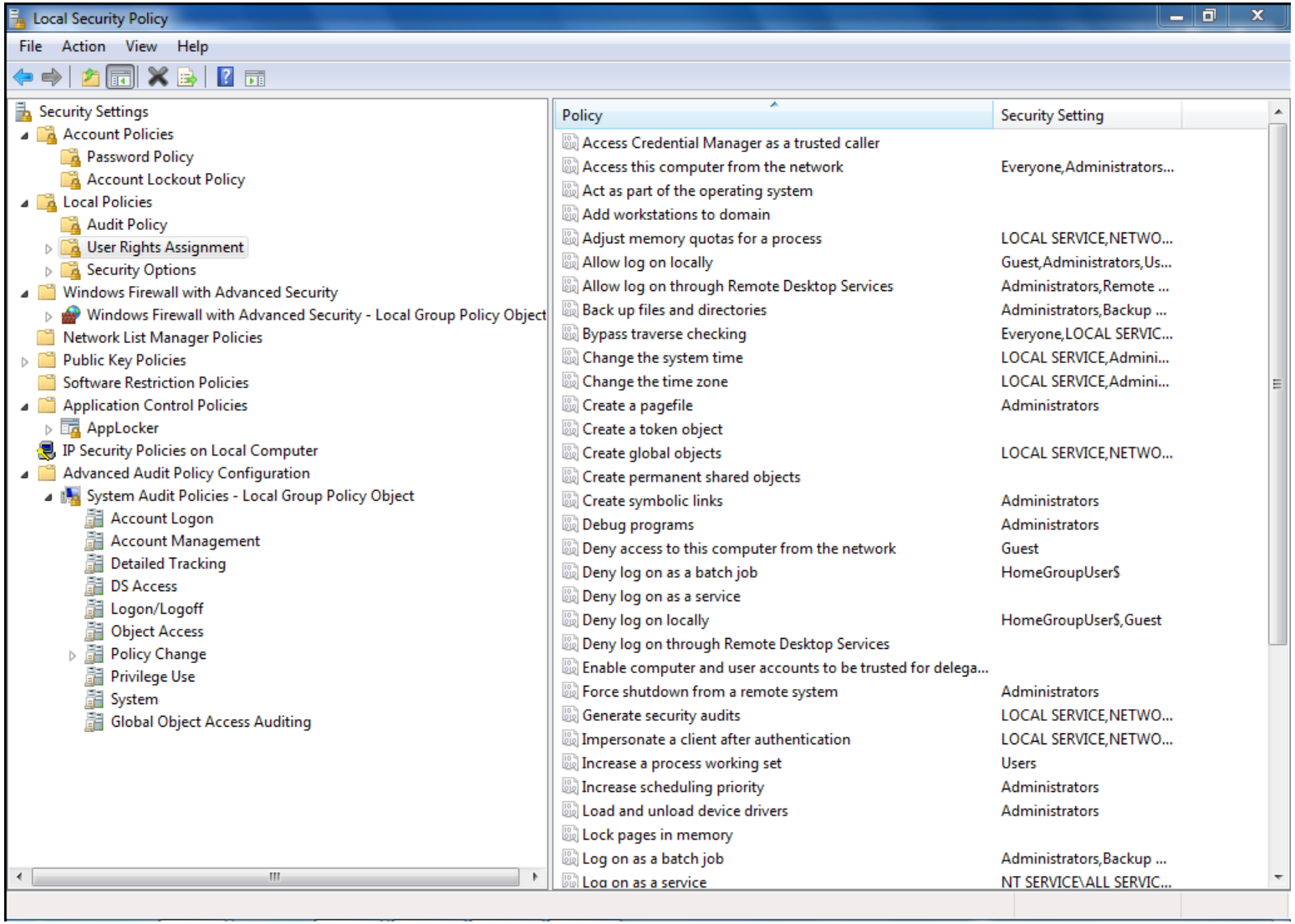
- Authentication Protocols
  - Kerberos
  - NTLMv2
- Authentication APIs
  - Win32 Security Support Provider Interface (SSPI)
  - .Net Framework System.Net.Security.NegotiateStream

Local Security Policy

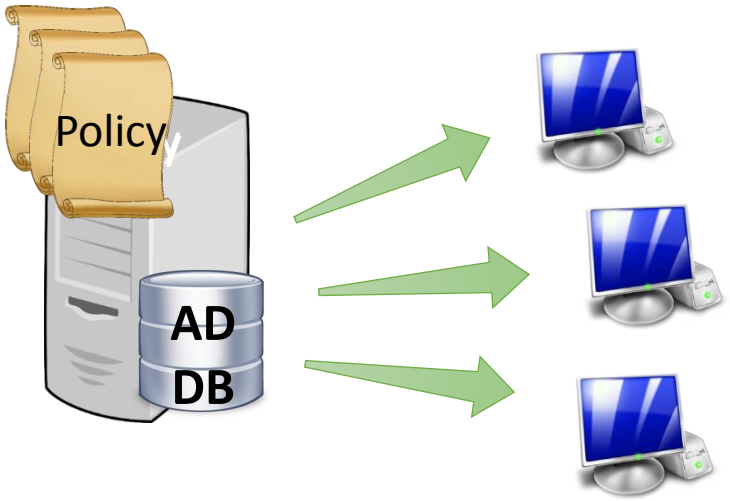
Local Security Policy

- Account Policies
  - Password Policy
  - Account Lockout Policy
- Local Polices
  - Audit Policy
  - Users rights assignment
  - Security Options
- Application Control Policies
- Other (Firewall/EFS/IPSec)

Local Security Policy (cont.)



Group Policy



- Manage user and computer settings
- Enforce IT policies
- Simplify administrative tasks
- Implement security settings
- Can be easily extended

Windows Services and Service Accounts

What is a Service?

- Long running program that operates in the background
- Designed to require no interactive user intervention
- Similar in concept to a Unix daemon
- Can be configured to start with the operating system

Service startup types

- Automatic
- Automatic (Delayed)
- Manual
- Disable

❖*Windows allows you to create dependencies if you want certain services to start before others.*

Service Control Manager (SCM)

- A special system process
- Responsible to interacts with Windows Service processes
  - Start
  - Stop
- Interacts through a well-defined API

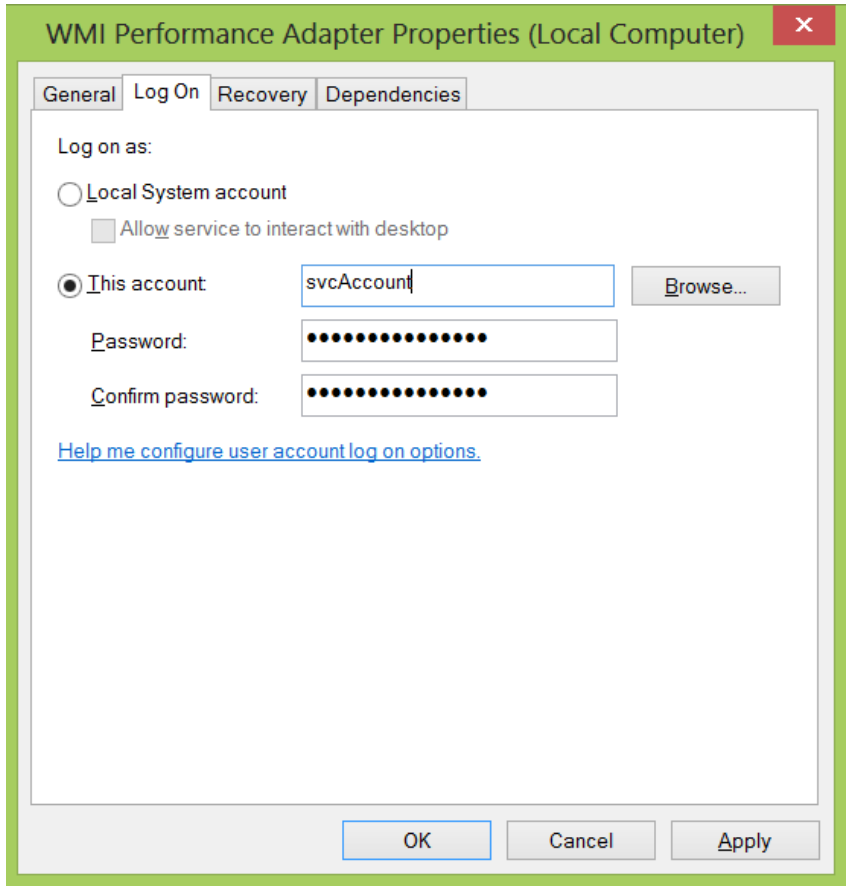
Develop Windows Service (C#)

```
public class MyService : System.ServiceProcess.ServiceBase
{
        public MyService()
        {
                InitializeComponent();
        }

        protected override void OnStart(string[] args)
        {
                //Add code that is executed on start
        }

        protected override void OnStop()
        {
                //Add code that is executed on stop
        }
}
```

Service Accounts

- Windows Services also runs from a context of account and also have access tokens
- Local or Domain
- Special Accounts
  - LocalSystem
  - LocalService
  - NetworkService

Log On Settings



User Account Control (UAC)

User Account Control

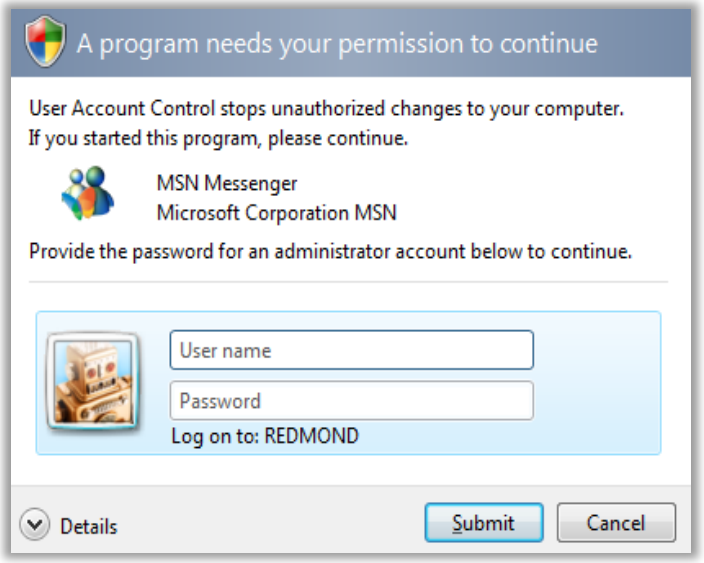
- How it works: When your consent is required to complete a task, UAC will prompt you with a dialog box
- Tasks that will trigger a UAC prompt include anything that will affect the integrity or security of the underlying system
  - This is a surprisingly long list of tasks
- UAC works slightly differently with standard user and administrator-class accounts
- UAC is working only for interactive sessions!

UAC Consent UI: Type 1

- Prompt: Windows needs your permission to continue
- Why you see this: You attempt to change a potentially dangerous system setting

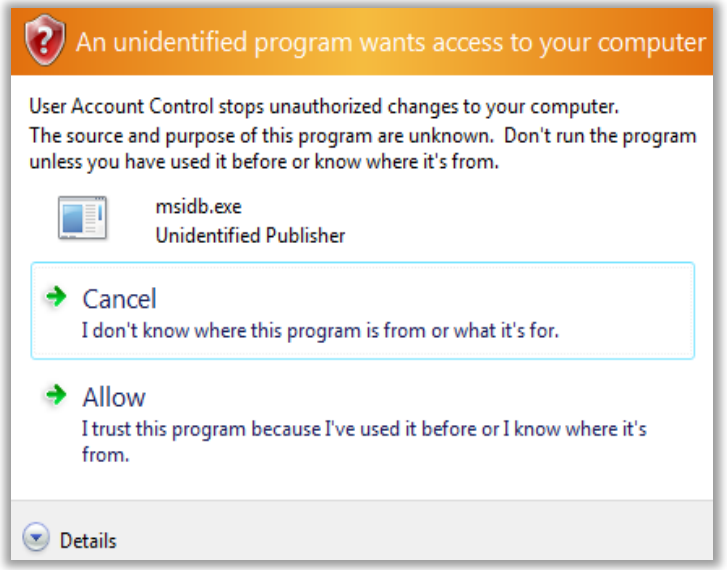
UAC Consent UI: Type 2

- Prompt: A program needs your permission to continue
- Why you see this: An external application with a valid digital signature is attempting to run with admin privileges



UAC Consent UI: Type 3

- Prompt: An unidentified program wants access to your computer
- Why you see this:
  - in external application without a valid digital signature is trying to run



UAC: What's really happening

- Administrator accounts logon with a mixed token
- Half of this mixed token is a standard user token
- The other half, the administrator token, is invoked only when required:
  - You can do so manually (run as)
  - Automatically (certain tasks in OS are tagged as requiring an admin token)

DPAPI

- OS service that provides confidentiality of data by using encryption
- Password-based data protection service
- Easy-to-use service
- Two Functions
  - CryptProtectData()
  - CryptUnprotectData()

Certificate Store

- Physical Stores
  - Local machine
  - Current user
- Logical Stores
  - Personal
  - Trusted Root Certification Authorities
  - Other

Where you can find resources?

- TechNet
  - https://technet.microsoft.com/
- MSDN
  - https://msdn.microsoft.com/
- Windows Protocols Specification
  - https://msdn.microsoft.com/en-us/library/jj712081.aspx

Windows Security Model