

Key Features of .NET Framework Security

- Fine control over security
- APIs allow easy authentication, authorization, and cryptographic functionality
- Protection from things like buffer overflows



1:16 / 3:02



Key Features of .NET Framework Security

- Type safety and security
- Principal
- Authentication

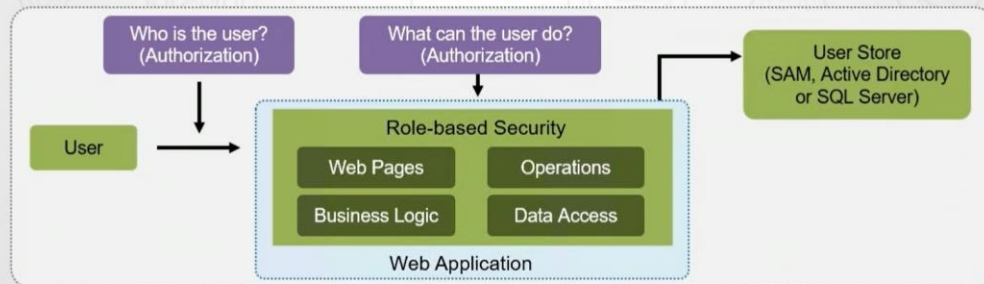


2:03 / 3:02



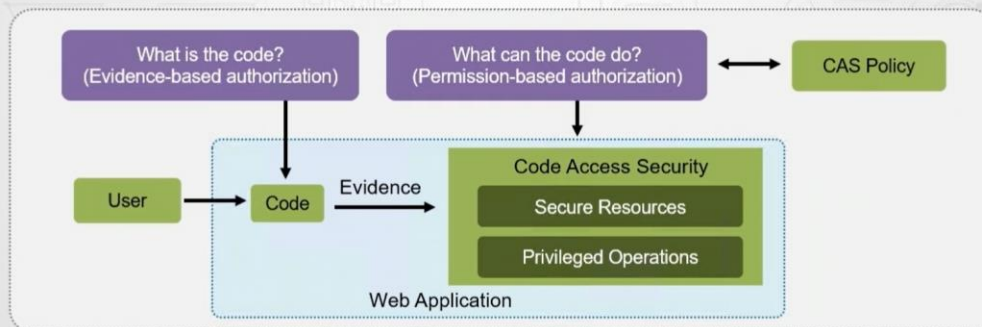
Role-based Security

- Permission based on user/group identity
- Authorizes user action in applications
- Works in conjunction with principal and identity objects



Code Access Security (CAS)

- Limits what the code can access
- Files, directories, settings, resources



Authentication in .NET

- Checks and verifies identity of principal
- Verifies credentials
- Can use .NET role-based security

WindowsPrincipal.IsInRole Object

To exit full screen, press **Esc**

- Checks if user is a member of the specified group (role)
- WindowsPrincipal.IsInRole methods
 - IsInRole(Int32)
 - Specified relative identifier (RID)
 - IsInRole(SecurityIdentifier)
 - Specified security identifier (SID)
 - IsInRole(String)
 - Specified name
 - IsInRole(WindowsBuiltInRole)
 - Specified WindowsBuiltInRole object



1:32 / 3:35



Windows Identity Foundation

- Access to application now based on claims
- Several templates available
- Trust between a claims-aware application and some security token service
- Facilitates conversion from NT-Token based to claims based requests
- Supports claims across multiple service boundaries (session claims are also possible)



1:07 / 4:49



Claims-based Identity

- Claims
- Security token
- Issuing authority
- Security token service

WIF Claims Programming Model

- Claims work with the F

Claim

```
+Claim Type: string  
+Value : string  
+ValueType : string  
+Issuer : string  
+OriginalIssuer : string  
+Properties : IDictionary <<string.|string>>  
+Subject : ClaimsIdentity
```



3:32 / 4:49

