ASP.NET Web API v2 Security Architecture

Dominick Baier
http://leastprivilege.com
@leastprivilege

pluralsight
hardcore dev and IT training

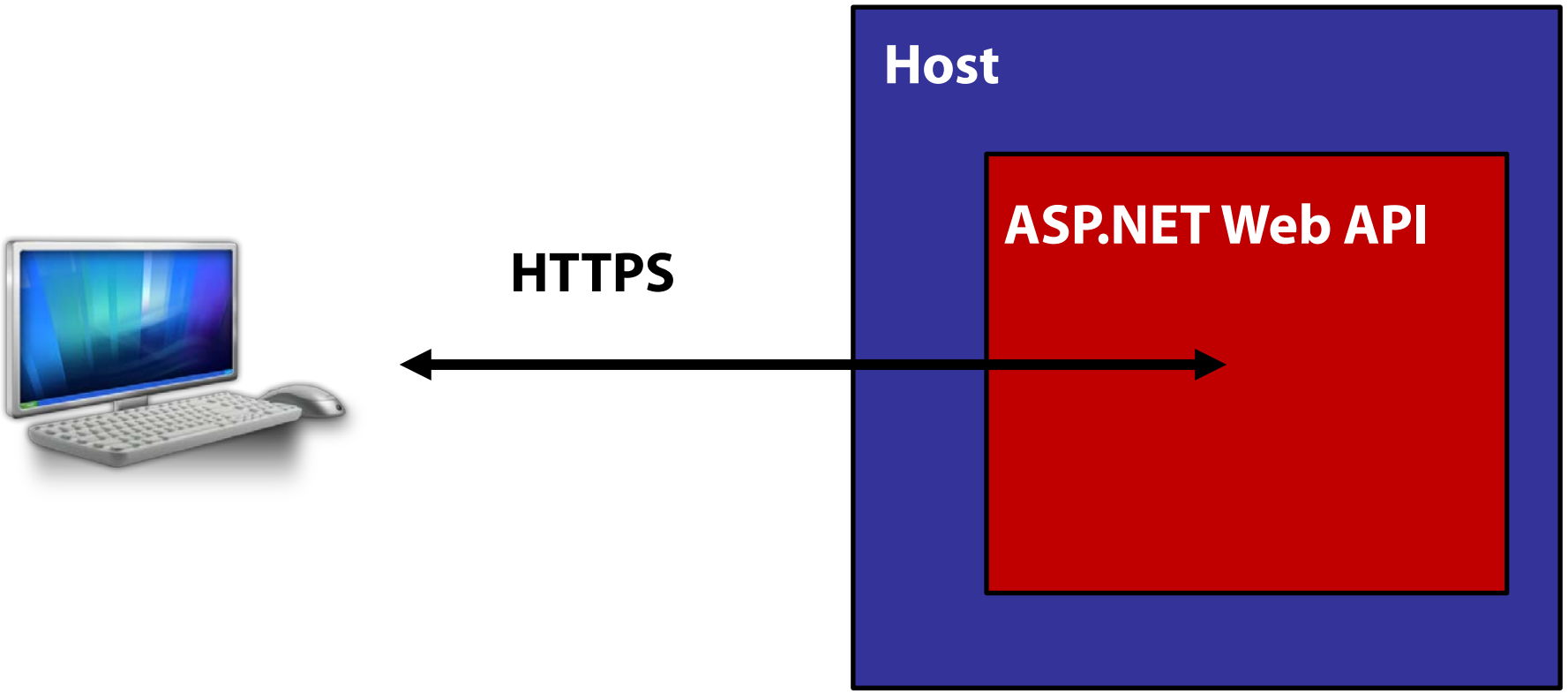
Agenda

- Overview of architecture

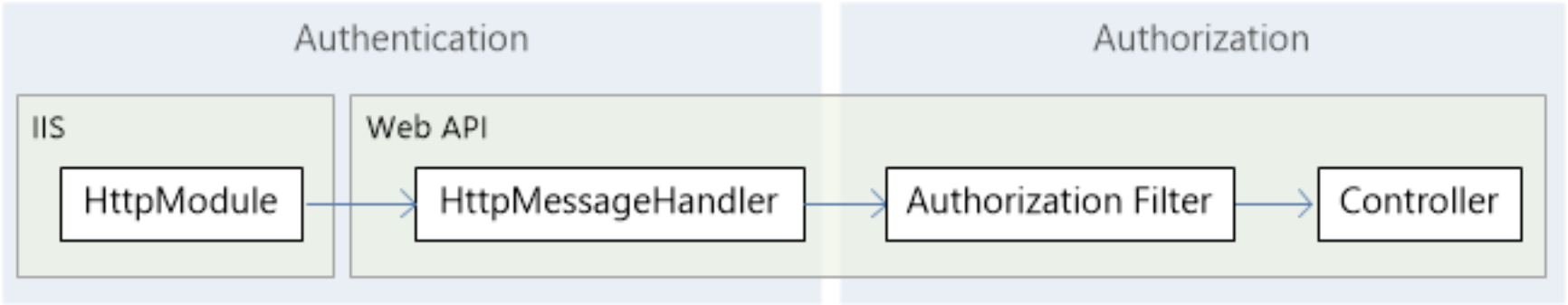
- Hosting
- Message handlers
- Authentication filters
- Authorization filters

- Accessing the client identity

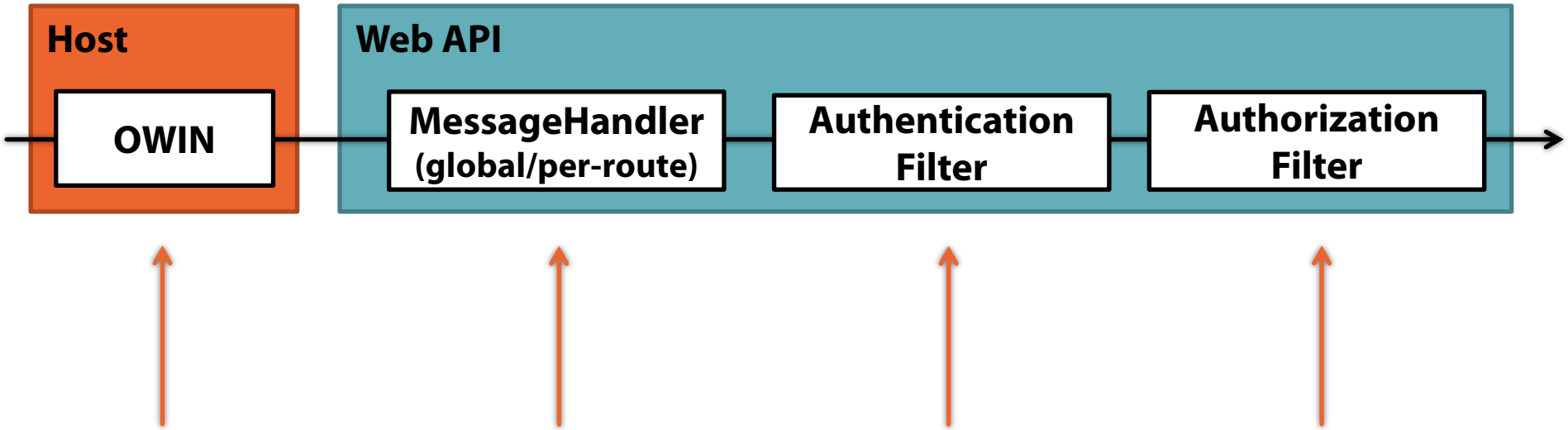
ASP.NET Web API: the big picture



Authentication & Authorization in Web API v1



The new Pipeline in Web API v2



Host/Framework independent concerns, e.g. authentication

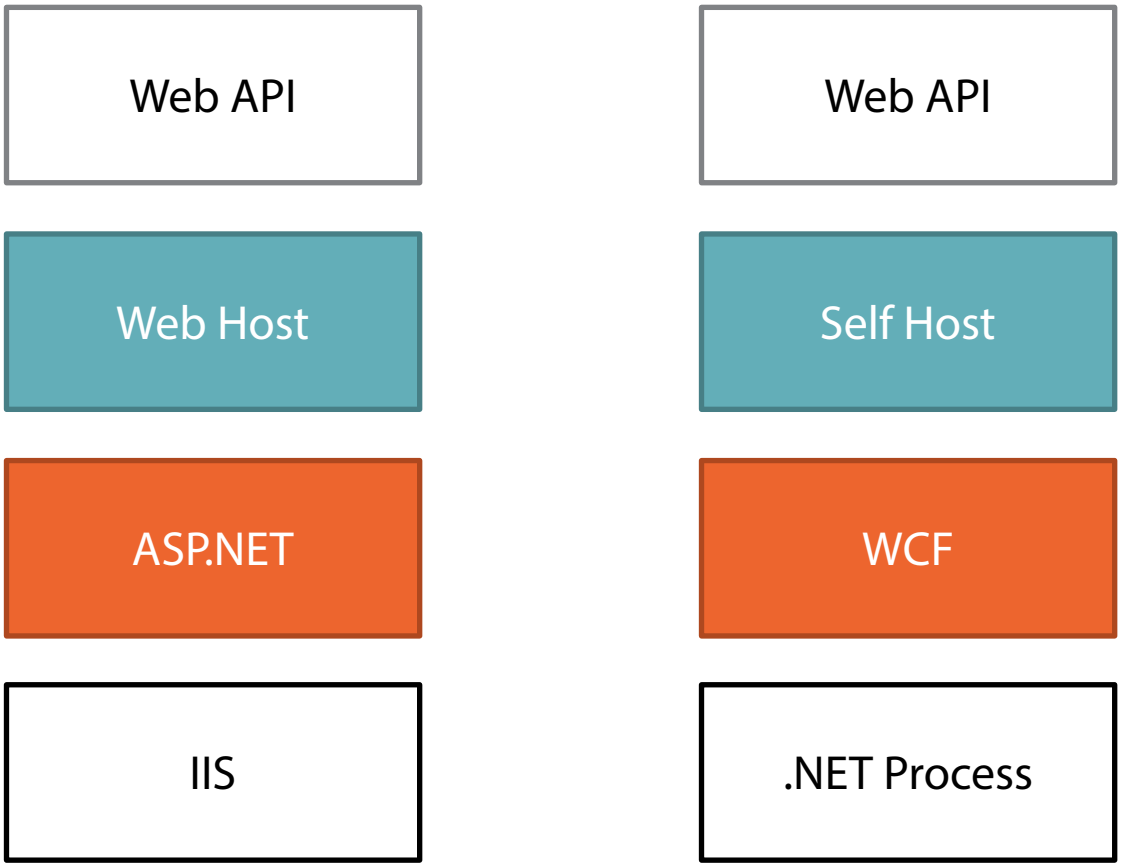
Web API cross-cutting concerns, e.g. CORS

Web API specific authentication

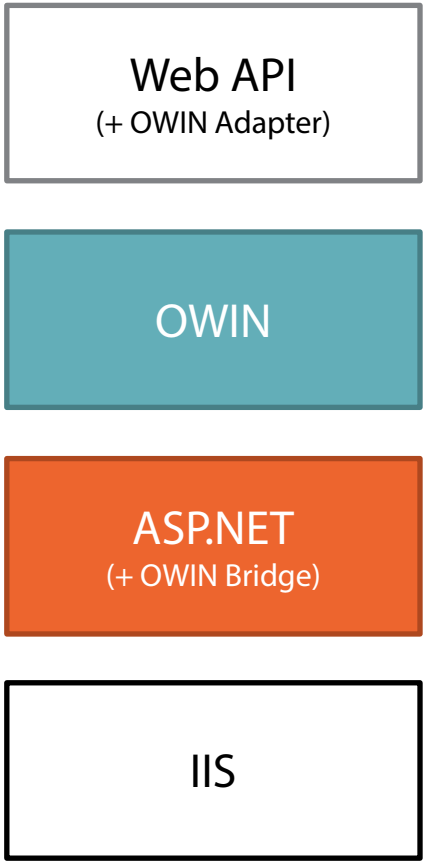
Authorization

http://www.asp.net/vnext/overview/owin-and-katana/an-overview-of-project-katana

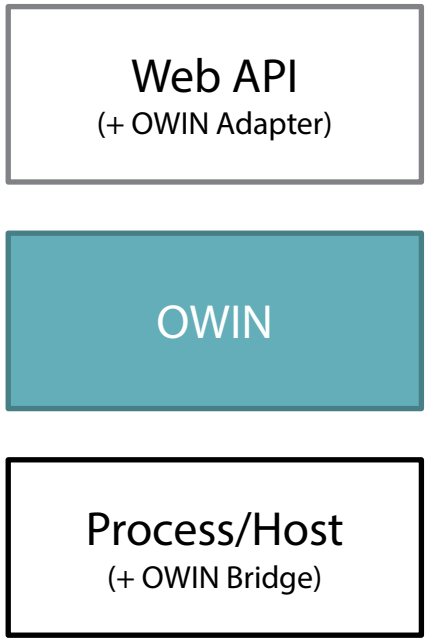
Classic hosting

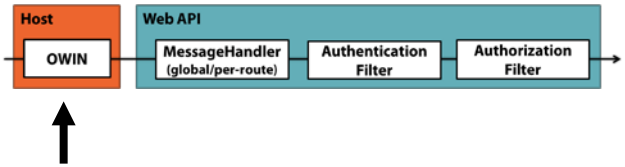


OWIN "System.Web" hosting



Pure OWIN hosting





OWIN Middleware

- Global

```
public class AuthenticationMiddleware
{
    private readonly Func<IDictionary<string, object>, Task> _next;

    public AuthenticationMiddleware(Func<IDictionary<string, object>, Task> next)
    {
        _next = next;
    }

    public async Task Invoke(IDictionary<string, object> env)
    {
        // inspect env and do credential validation, then set principal

        env["server.User"] = CreatePrincipal();
        await _next(env);
    }
}
```

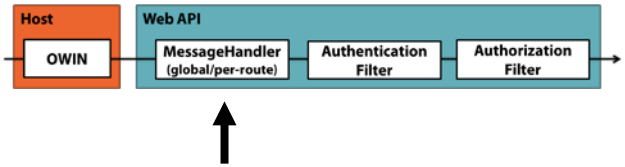
Katana Authentication Middleware

```
public class Startup
{
    public void Configuration(IAppBuilder app)
    {
        app.UseCookieAuthentication(new CookieAuthenticationOptions
            {
                AuthenticationType = "Cookies",
                // more options
            });

        app.UseGoogleAuthentication(new GoogleAuthenticationOptions
            {
                AuthenticationType = "Google",
                // more options
            });

        app.UseOAuthBearerAuthentication(new OAuthBearerAuthenticationOptions
            {
                AuthenticationType = "Bearer"
                // more options
            });
    }
}
```

ything

ng

editing)
Note: Warning will not appear during Slide Show view.



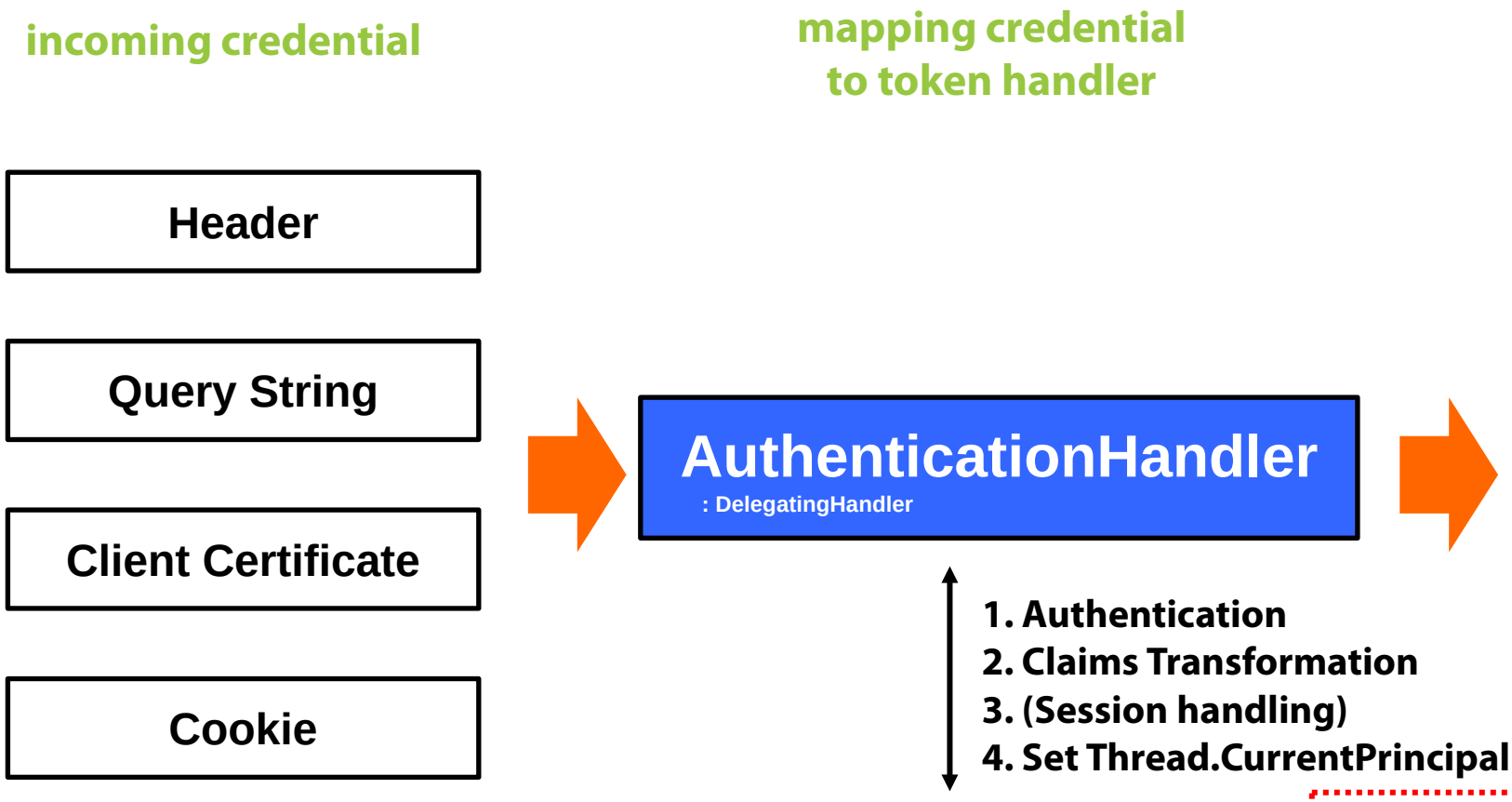
MessageHandler

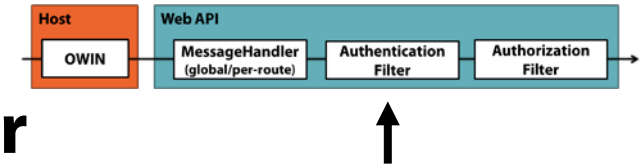
- Web API, global or per-route

```
public class MyHandler : DelegatingHandler
{
    protected async override Task<HttpResponseMessage> SendAsync(
      HttpRequestMessage request, CancellationToken cancellationToken)
    {
        // inspect request

        var response = await base.SendAsync(request, cancellationToken);

        // inspect response
        return response;
    }
}
```

Thinktecture AuthenticationHandler

http://thinktecture.github.com/Thinktecture.IdentityModel/



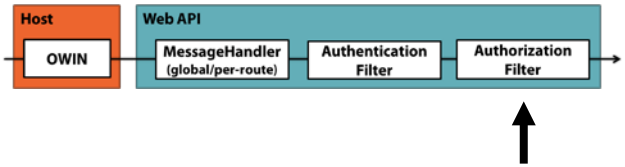


Authentication filter

WebApiConfig.cs

```
config.Filters.Add(new HostAuthenticationFilter("Bearer"));
```

```
[HostAuthentication("Bearer")]
public class TestController : ApiController
{
    [HostAuthentication("Google")]
    public HttpResponseMessage Get()
    { }

    [OverrideAuthentication]
    [HostAuthentication("Cookies")]
    public HttpResponseMessage Delete()
    { }
}
```



Authorization filter

- **Determines if a resource needs authentication**
  - *[AllowAnonymous]* to skip authorization for an action
  - emits the 401 status code, if unsuccessful

```
// minimum requirement is successful authentication
[Authorize]
public DataController : ApiController
{
    [AllowAnonymous]
    public Data Get()
    { … }

    [Authorize(Role = "Foo")]
    public HttpResponseMessage Delete(int id)
    { … }
}
```

Accessing the client identity

- **Previous versions of Web API used *Thread.CurrentPrincipal* to access client identity**
  - *ApiController.User* was a shortcut to T.CP

- **v2 uses a new concept: *RequestContext***
  - hangs off the *HttpRequestMessage*
  - *ApiController.User* is now a shortcut to the request context
  - could be *null*

Summary

- **Web API security extensibility is a pipeline**
  - Katana middleware
  - message handlers (not encouraged anymore)
  - authentication filters
  - authorization filters
- **Avoid host (e.g. IIS) specific dependencies**

- **HttpRequestMessage.GetRequestContext().Principal**
  - is the one stop shopping for client identity

Resources

- **PluralSight:**
  - Scott Allen - MVC5 Fundamentals (OWIN and Katana)
  - Dominick Baier - Introduction to Identity & Access Control in .NET 4.5