# Cross Site Request Forgery (CSRF) Prevention for ASP.NET Core and ASP.NET Applications

## UNDERSTANDING AND MITIGATING CSRF

**Roland Guijt**
MICROSOFT MVP, CONSULTANT, AUTHOR AND SPEAKER

@rolandguijt  rolandguijt.com

# Module Overview

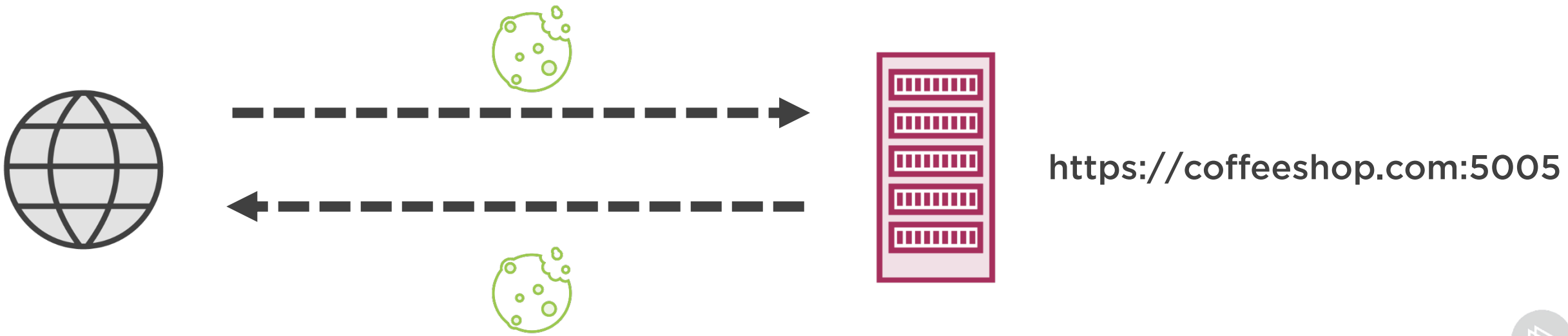
What is CSRF?

SameSite cookies

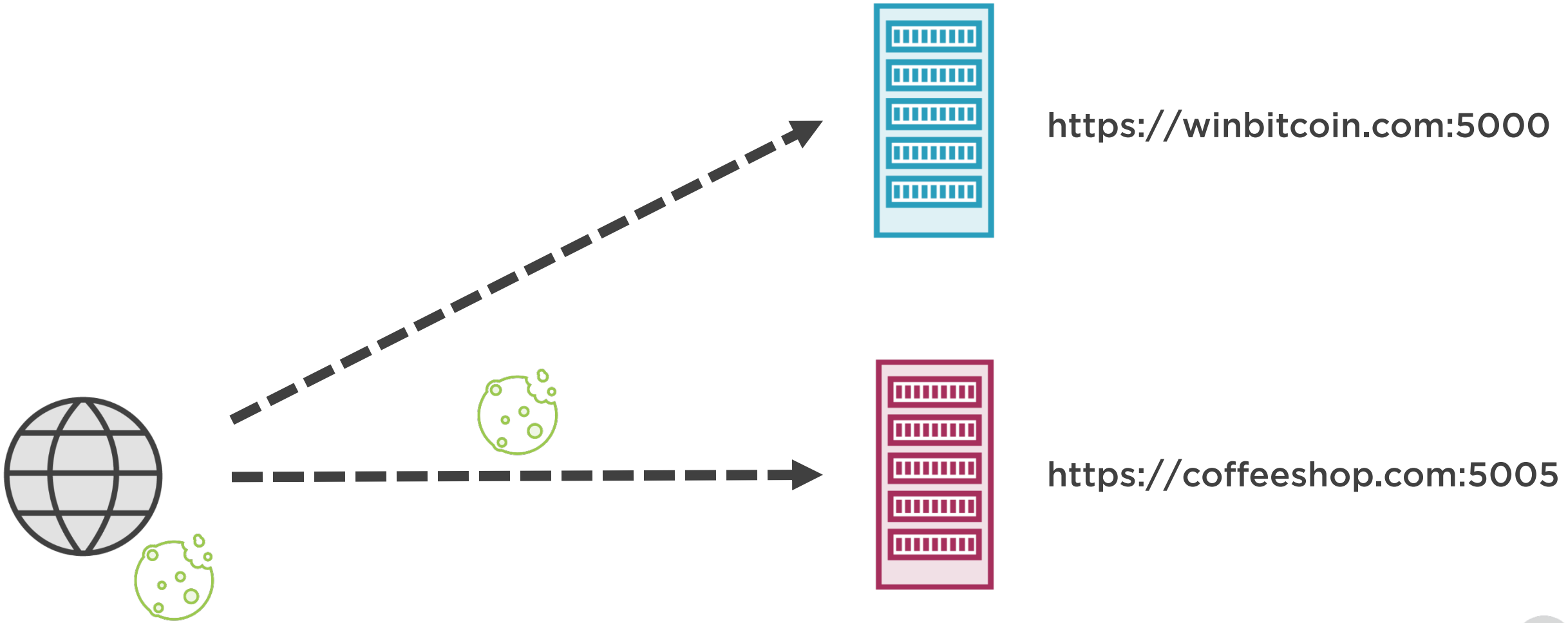
Anti forgery tokens

Dos and don’ts

# Cookies



# Cookies and CSRF



# Attack Requirements

- User must have visited attacked site
- User must be lured to the site of the attacker
- Attackers look for profit
- They are after your identity cookie

# Authentication and Authorization in ASP.NET Core

Setting SameSite limits the types of requests that are possible with the cookie

# SameSite Options

**Strict**

**Lax**

**None**

# SameSite and Browser Defaults

**Browsers (will) default to Lax**

**Omit SameSite => SameSite == Lax**

**None disables SameSite but requires secure**

# Problem solved?

https://caniuse.com/#feat=same-site-cookie-attribute

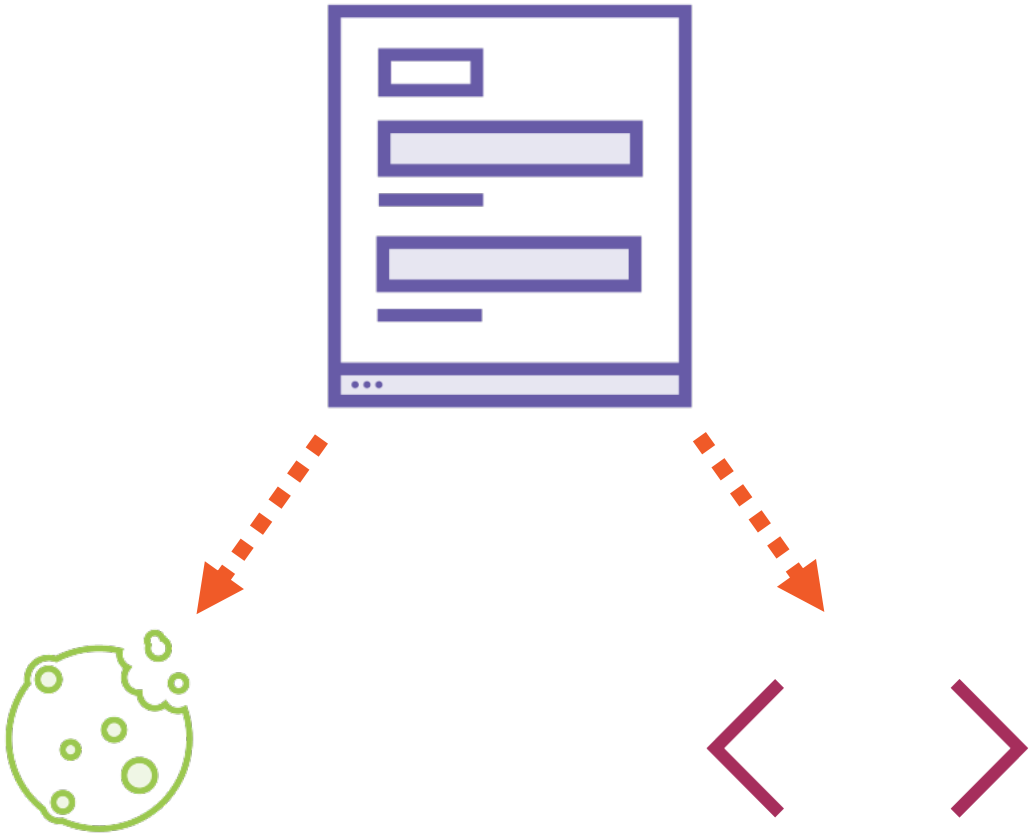
# ASP.NET SameSite Support

.NET Framework 4.7.2 or higher

or

.NET Core 2.1 or higher

# Anti Forgery Tokens



# Anti Forgery Tokens

**Difficult for attacker to generate encrypted valid code for hidden input**

**Key are known to the attacked web application only**

**Difficult for attacker to set a cookie for the attacked site**

# Best Practices

**Use SameSite cookies**

- strict mode if possible
- lax mode as a minimum

**Consider ruling out older browsers**

**Use anti forgery tokens**

**Always use the POST HTTP method with HTML forms**

**Be careful with supporting command requests**

```
<form method="get" action="coffee.com">
```

4sh.nl/ajaxantiforgerytokens

# Best Practices (continued)

**Never support GET requests that change data or state**

**Instead use POST requests**

```
document.cookie = "loyaltyNumber=1234; SameSite=Lax";
```

# Additional Practices for SPAs

- Setup the API in a separate (sub) domain
- Enable CORS using specific rules
- Don't rely on CORS alone
- Use built-in JavaScript support for AJAX requests or libraries build on top

# Summary

CSRF is an attack that causes the victim to carry out an action unintentionally exploiting cookies

SameSite cookies solve most of the problem

Anti forgery tokens

Best practices and additional attack vectors