Kako prilagoditi SameSite atribut kolačića

SameSite atribut kolačića (eng. SameSite cookie attribute) je dio zaglavlja Set-Cookie u HTTP odgovorima. Atribut SameSite određuje da li će kolačić biti dostupan samo u "first-party" ili i u "third-party" kontekstu. Preglednici Chrome, Firefox, Edge i drugi mijenjat će svoje zadano ponašanje u skladu s IETF-ovim prijedlogom "Incrementally Better Cookies" tako da:

- kolačići bez atributa SameSite tretirat će se kao "SameSite=Lax", što znači da će zadano ponašanje biti ograničavanje kolačića samo na kontekst "first-party".
- kolačići za upotrebu na različitim web lokacijama moraju navesti "SameSite=None; Secure", što omogućuje uključenje u kontekst "third-party".

Ovo postaje standardno ponašanje u Chromeu 80, planirano za stabilno izdanje u veljači 2020.

Ograničavanje kolačića samo na kontekst "first-party" onemogućuje ispravno funkcioniranje AAI (SAML) autentikacije, jer HTTP POST zahtjev sa vanjske strane (login.aaiedu.hr) na stranu koja je zahtjevala operaciju (SP) neće poslati potrebne kolačiće na SP.

Na primjer, zahtjev sa https://login.aaiedu.hr na https://primjer-sp.hr /login.ashx je u zaglavlju HTTP zahtjeva vratio:

```
Set-Cookie: ASP.
NET_SessionId=zqtgndqojc0regka43qhkkxt; path=/;
HttpOnly; SameSite=Lax
```

Dakle, zbog atributa "SameSite=Lax", preglednik neće uključiti kolačić "ASP.NET_SessionId" u naknadnim zahtjevima prema https://primjer-sp. hr/login.ashx, što onemogućuje autentikaciju. Općenito govoreći, takva postavka onemogućuje preglednik da pošalje ispravan kolačić SPu nakon uspješne autentikacije na SSO servisu, zbog čega SP ne zna da korisnik ima uspostavljenu sjednicu. Da bi se kolačić uključivao u HTTP zahtjevima potrebno je postaviti ove atribute prilikom postavljanja kolačića: "SameSite=None; Secure".

Linkovi s detaljnijim objašnjenjima:

- https://web.dev/samesite-cookies-explained/
- https://web.dev/samesite-cookie-recipes/

Omogućavanje kolačića u kontekstu "third-party"

Pošto je SameSite atribut relativna novost, većina web servera / aplikacija ga u trenutku pisanja ovog teksta ne postavlja. Ponašanje preglednika je, za sada, da omogućavaju kontekst "third-party" ako nema postavljenog atributa "SameSite" na vrijednost "Lax" ili "Strict", ali takvo ponašanje prestaje u veljači 2020. Novo ponašanje preglednika je da ako nema postavljenog atributa "SameSite", podrazumija se vrijednost "SameSite=Lax". Takva postavka onemogućuje preglednik da pošalje ispravan kolačić SPu nakon uspješne autentikacije na SSO servisu zbog čega SP ne zna da korisnik ima uspostavljenu sjednicu.

Google je pripremio primjere postavljanja atributa SameSite za nekoliko jezika / platformi: https://github.com/GoogleChromeLabs/samesite-examples

Također, u nastavku slijedi nekoliko primjera konfiguracije.

IIS

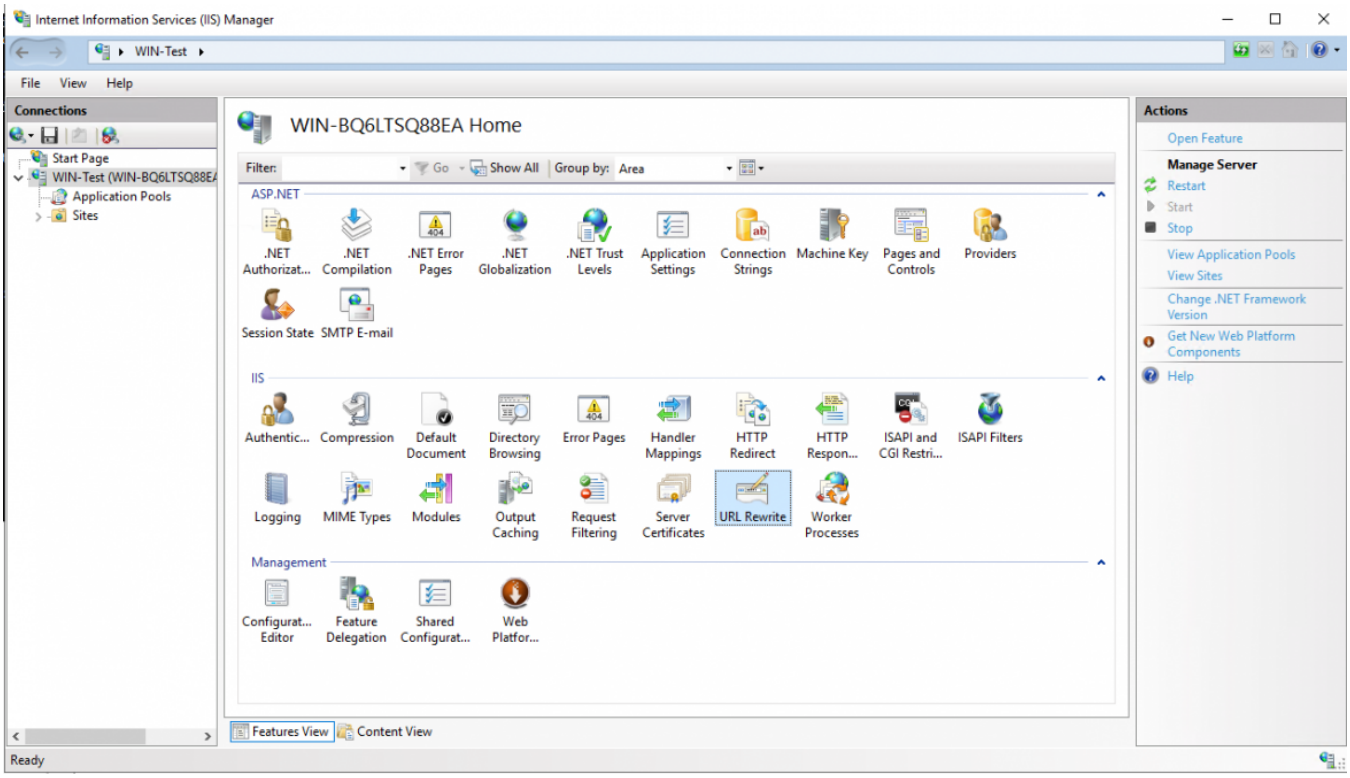
Primjer kako postaviti atribut "SameSite" na razini IIS servera dostupan je na https://www.petefreitag.com/item/850.cfm.

Potrebno je obratiti pažnju da je u uputi navedeno kako postaviti "SameSite=Lax", a mi trebamo postaviti "SameSite=None; Secure".

Prije postavljanja pravila za atribut SameSite potrebno je:

- ako već nije instaliran, instalirati URL Rewrite modul za IIS: https://www.iis.net/downloads/microsoft/url-rewrite
- ponovno pokrenuti IIS

Nakon toga pojavljue se modul "URL Rewrite" kao dostupna opcija:



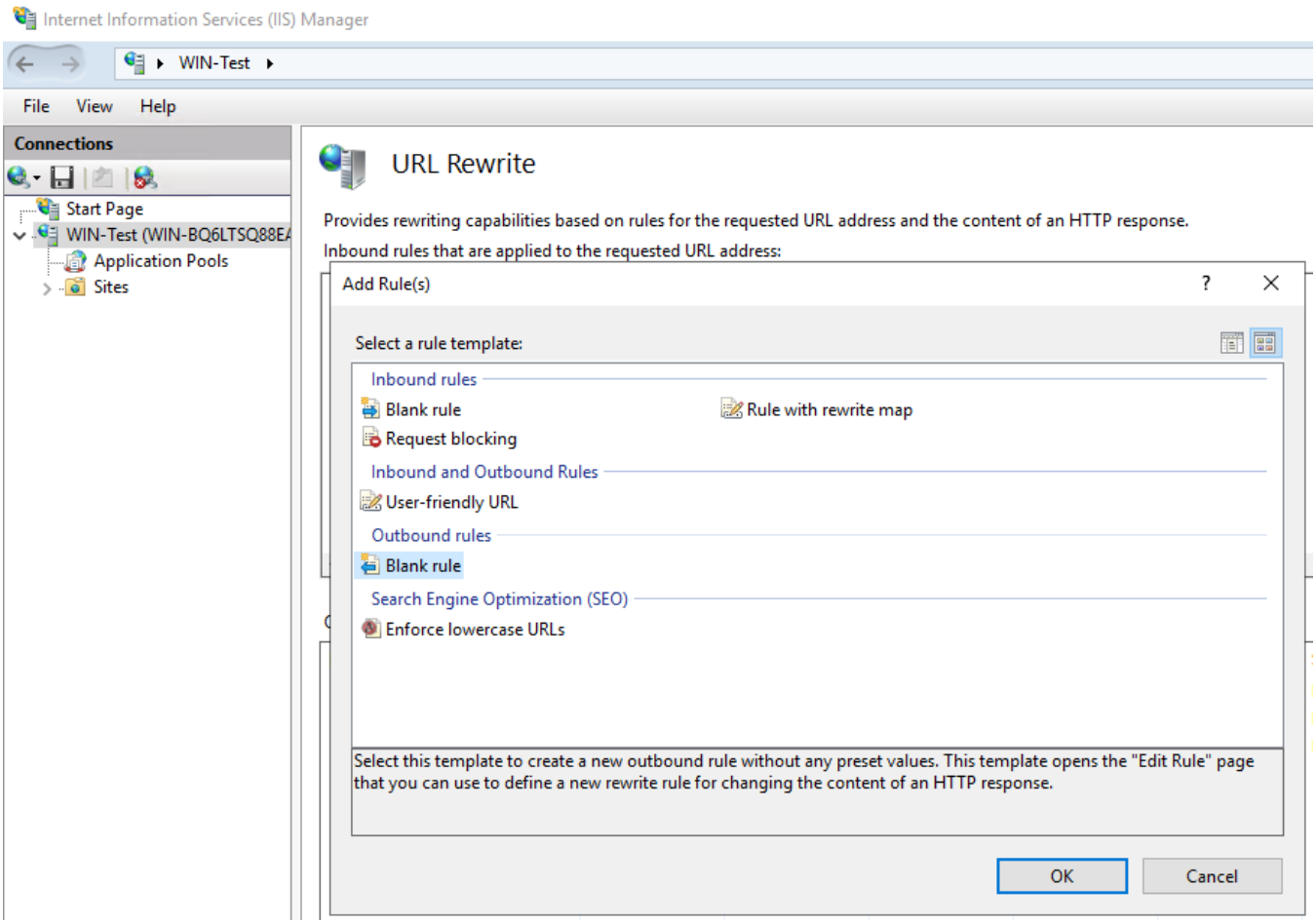
Ovisno na kojoj razini je potrebno postaviti atribut SameSite, potrebno je odabrati čitav server ili pojedino web sjedište, pa otvoriti "URL Rewrite" modul. Ako se odabere čitav server, pravila će se primjenjivati na sva web sjedišta na serveru. Ako se odabere pojedino web sjedište, pravila će se primjenjivati samo za odabrano web sjedište.

Postavljanje pravila za sve kolačiće

Postavljanje i normaliziranje atributa SameSite za sve kolačiće koje će isporučivati server (ili pojedino web sjedište) može se napraviti kroz definiranje nekoliko pravila. Ono što je potrebno osigurati jest da svi kolačići imaju atribute "SameSite=None; Secure". Pošto različiti kolačići mogu imati različitie atribute, te već mogu imati postavljen atribut SameSite (npr. na SameSite=Lax), potrebno je normalizirati kolačiće na način da osiguramo da nemaju definiran atribut SameSite=Lax|Strict i da uvijek imaju atribut secure. To je moguće napraviti kroz tri koraka:

- makni atribut "Secure" ako postoji (dodat ćemo ga sami)
- makni atribut "SameSite=Lax" ili "SameSite=Strict" ako postoji (jer ćemo postaviti atribut "SameSite=None")
- postavi atribute "SameSite=None; Secure"

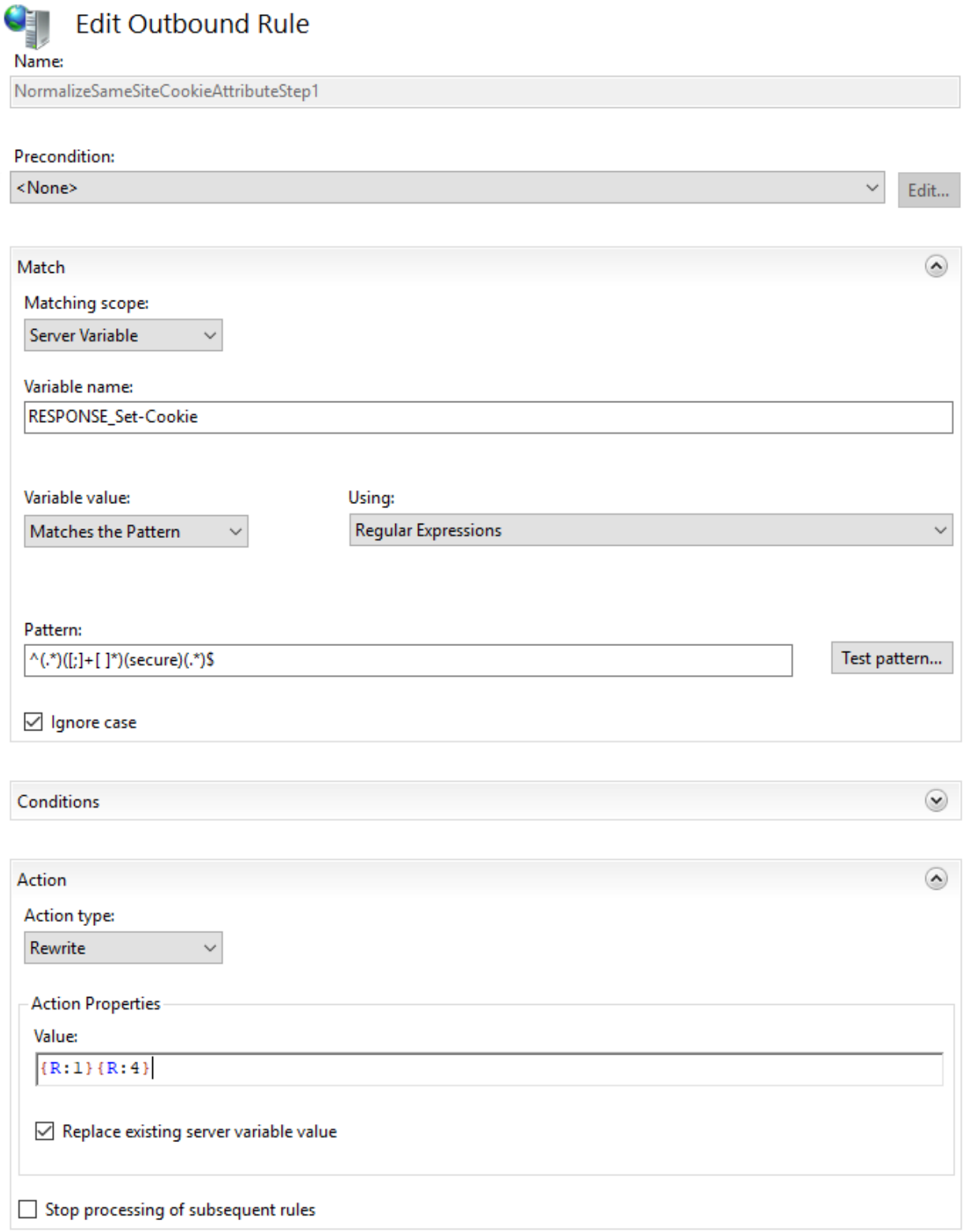
Krenimo s prvim korakom. Potrebno je otvoriti "URL Rewrite", pa odabrati opciju "Add Rule" > "Outbound rules" > "Blank rule".



U prozoru koji se otvori potrebno je definirati sljedeće:

- "Name" kao proizvoljan tekst, npr. "NormalizeSameSiteCookieAttributeStep1"
- u odjeljku "Match" postaviti sljedeće opcije:
    - za "Matching scope" odabrati "Server Variable"
    - za "Variable name" upisati: RESPONSE_Set-Cookie
    - za "Variable value" odabrati "Matches the Pattern"
    - za "Using" odabrati "Regular Expressions"
    - za "Pattern" upisati: ^(.*)([;]+[ ]*)(secure)(.*)$
    - za "Ignore case" postaviti kvačicu (odabrati ju)
- u djeljku "Action" postaviti sljedeće opcije:
    - za "Action type" odabrati "Rewrite"
    - za "Action Properties", "Value" upisati: {R:1}{R:4}
    - za "Replace existing server variable value" postaviti kvačicu (odabrati ju)

Čitava definicija prvog pravila sada izgleda ovako:

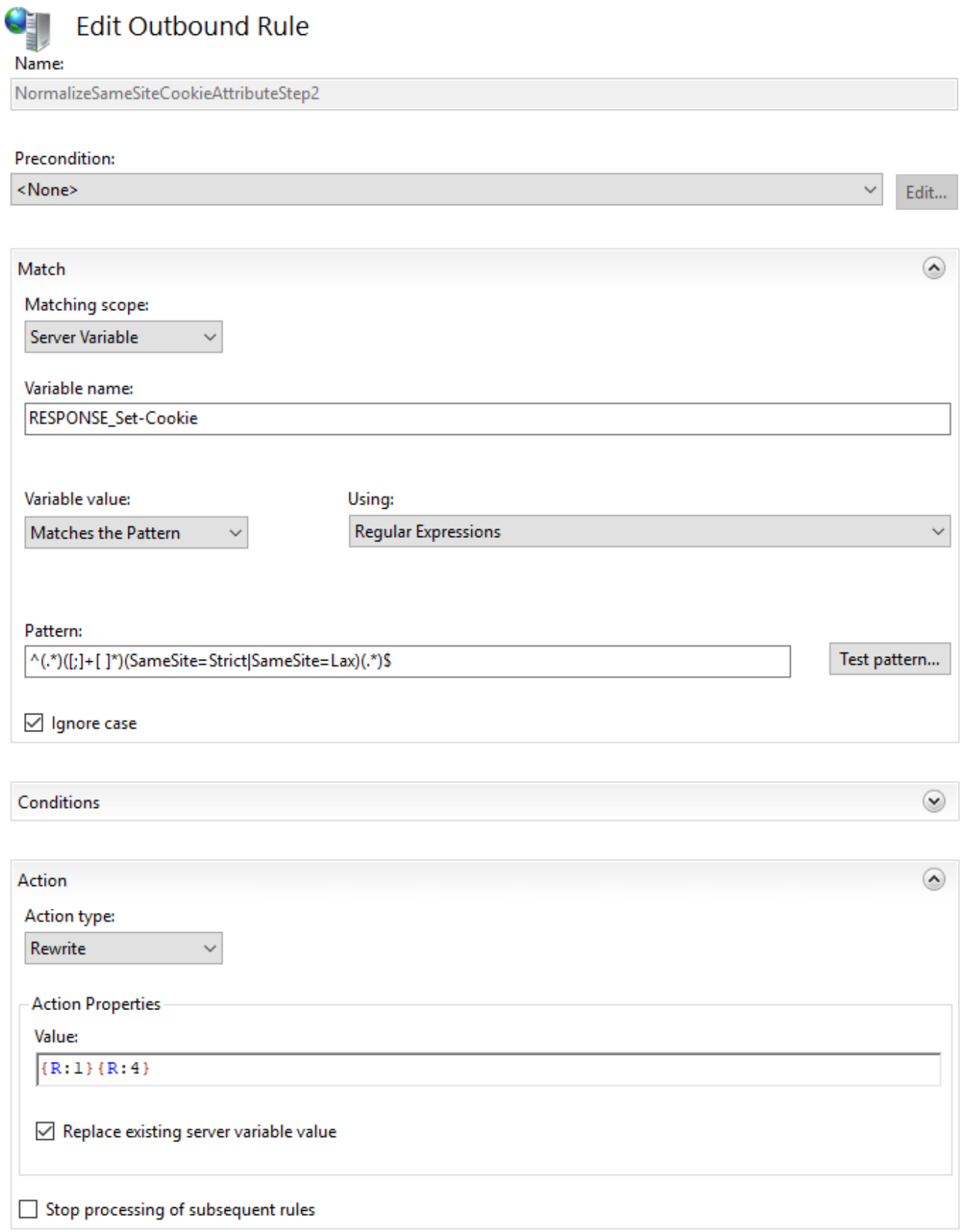


Ovim pravilom u kolačiću brišemo atribut "Secure" (ako postoji), dok sve ostale vrijednosti u kolačiću ostaju.

Za sljedeći korak definirat ćemo novo pravilo, koje će se od prvog razlikovati u sljedećim opcijama:

- "Name": NormalizeSameSiteCookieAttributeStep2
- u odjeljku "Match"
    - za "Pattern" upisati: ^(.*)([;]+[ ]*)(SameSite=Strict|SameSite=Lax)(.*)$

Čitava definicija drugog pravila sada izgleda ovako:

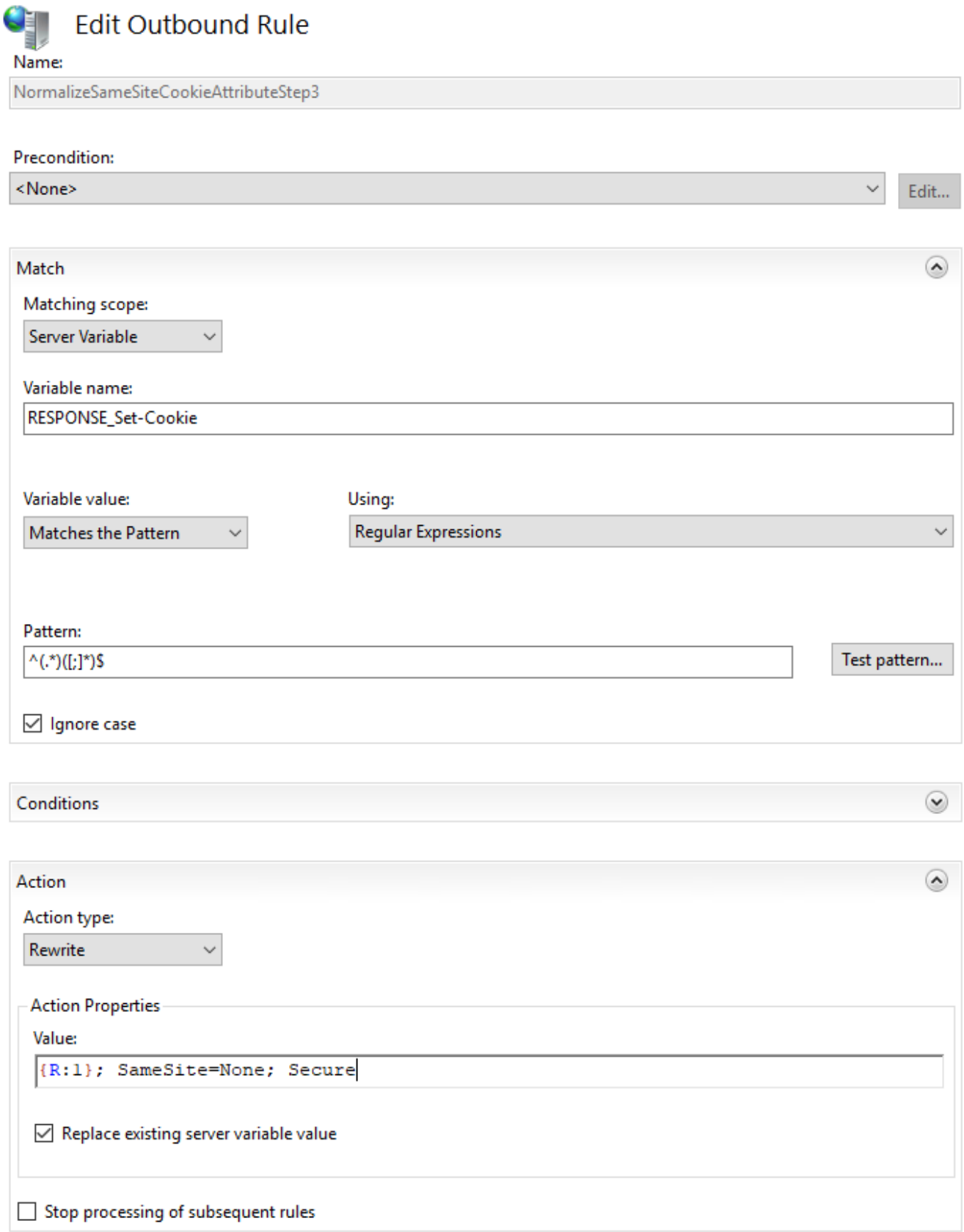


Ovim pravilom u kolačiću brišemo atribut "SameSite=Strict" ili "SameSite=Lax" (ako postoji), dok sve ostale vrijednosti u kolačiću ostaju.

Za sljedeći korak definirat ćemo novo pravilo, koje će se od prva dva razlikovati u sljedećim opcijama:

- "Name": NormalizeSameSiteCookieAttributeStep3
- u odjeljku "Match"
    - za "Pattern" upisati: ^(.*)([;]*)$
- u odjeljku "Action"
    - za "Action Properties", "Value" upisati: {R:1}; SameSite=None; Secure

Čitava definicija trećeg pravila sada izgleda ovako:



Sumarni prikaz sva tri pravila sada izgleda ovako:

Outbound rules that are applied to the headers or the content of an HTTP response:

Name	Input	Match	Pattern	Action Type	Action Value	Stop Processing	Entry Type
NormalizeSameSiteCookieAttributeStep1	RESPONSE_Set-Cookie	Matches	^(.*)([;]+[ ]*)(secure)(.*)$	Rewrite	{R:1}{R:4}	False	Local
NormalizeSameSiteCookieAttributeStep2	RESPONSE_Set-Cookie	Matches	^(.*)([;]+[ ]*)(SameSite=Strict|SameSite=Lax)(.*)$	Rewrite	{R:1}{R:4}	False	Local
NormalizeSameSiteCookieAttributeStep3	RESPONSE_Set-Cookie	Matches	^(.*)([;]*)$	Rewrite	{R:1}; SameSite=None; Secure	False	Local

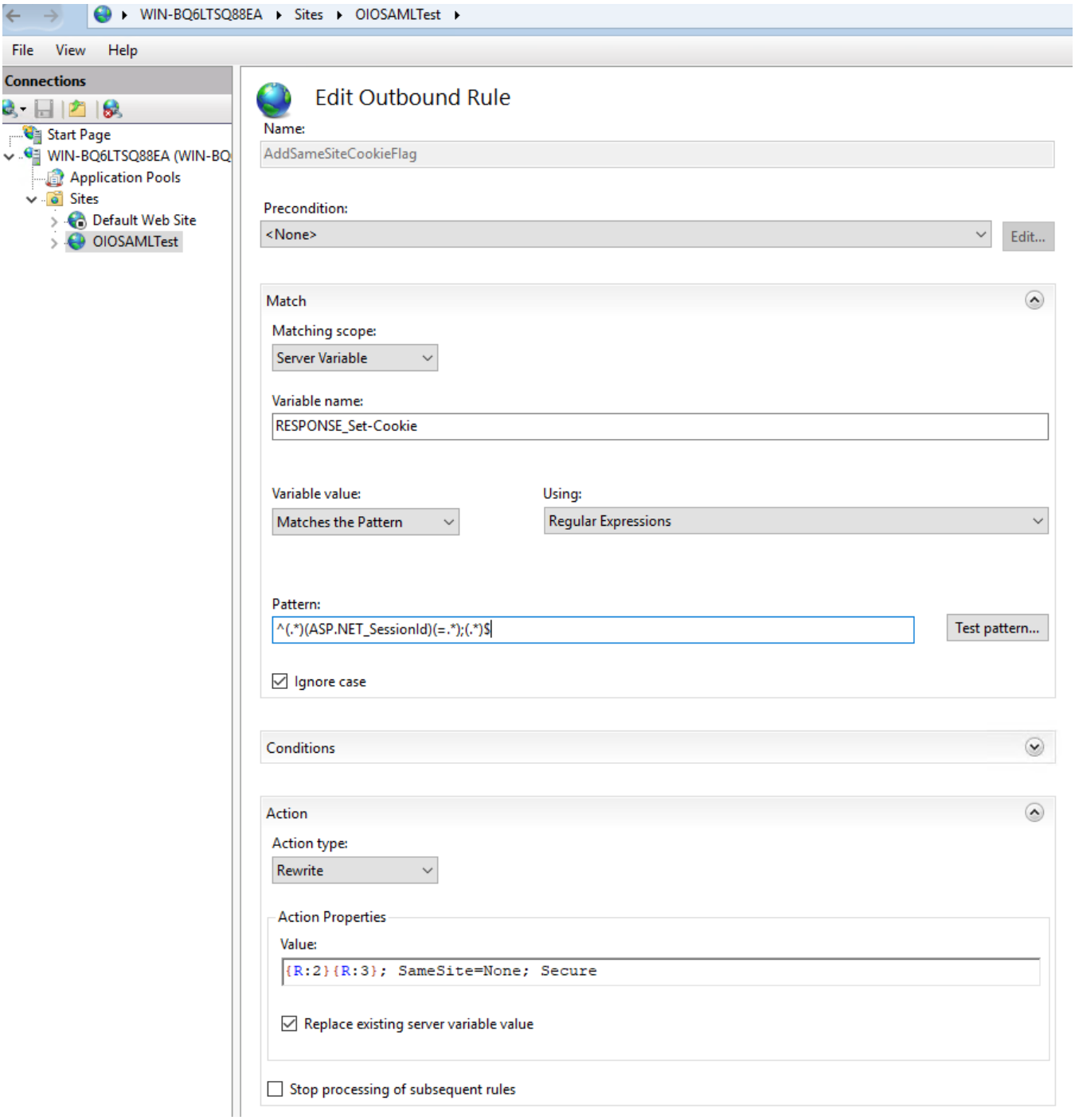
Iako se na ovaj način osigurava da svi kolačići imaju atribute "SameSite=None; Secure", čime je omogućeno funcioniranje SAML autentikacije, potrebno je imati na umu da se time efektivno zaobilazi IETF inicijativa "Incrementally Better Cookies". Ako znate koji točno kolačići služe u procesu autentikacije, možete omogućiti atribute "SameSite=None; Secure" na samo određenim kolačićima.

Postavljanje pravila na samo određenom kolačiću

Recimo da imamo sljedeći HTTP header kojeg vraća davatelj usluge (Service Provider - SP), a koji se na SPu koristi tijekom autentikacije korisnika (naznačuje korisnikovu sesiju): "Set-Cookie: ASP.NET_SessionId=zqtgndqojc0regka43qhkkxt; path=/; HttpOnly; SameSite=Lax".

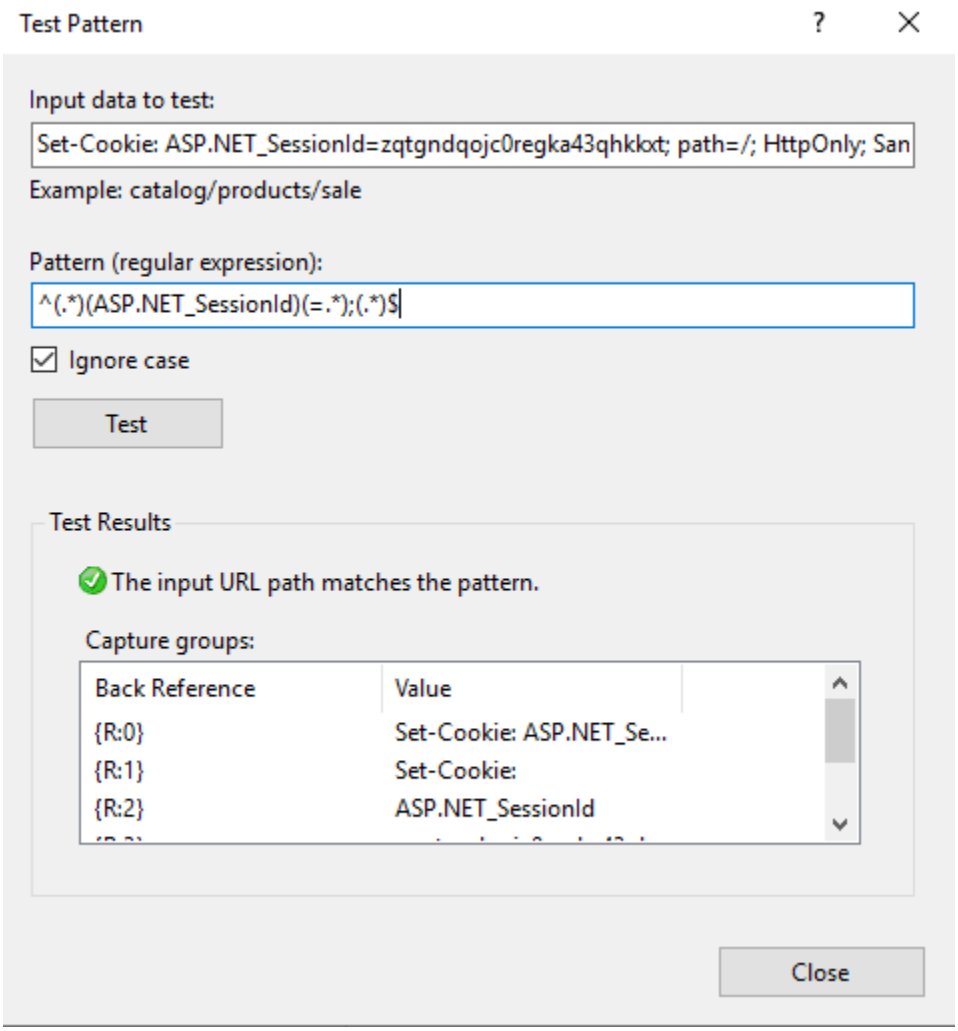
Iz primjera vidimo da se koristi atribut "SameSite=Lax", što znači da za taj kolačić nije omogućen kontekst "third-party", što efektivno onemogućuje AAI SAML autentikaciju.

Primjera radi, za njega možemo definirati ovakav "pattern": "^(.*)(ASP.NET_SessionId)(=.*);(.*)$", a kao "Action Properties Value" možemo postaviti "{R:2}{R:3}; SameSite=None; Secure" (sve bez navodnika), ovako:



Dakle, "pattern" je uvijek potrebno prilagoditi određenom nazivu kolačića, a "Action Properties Value" je potrebno prilagoditi obzirom na sadržaj kolačića.

"Pattern" se može relativno lako testirati pomoću "Test Pattern" funkcionalnosti, pa shodno tome definirati i "Action Properties Value". Primjer testiranja patterna na realnom headeru:



Istu stvar je potrebno napraviti na svim kolačićima koji su potrebni u procesu autentikacije.

Apache

Slično kao i u primjeru za IIS, pomoću modula "mod_haders(link is external)" možemo mijenjati HTTP zaglavlja na web poslužitelju Apache. Na primjer, možemo postaviti sljedeća pravila:

```
Header edit Set-Cookie "^(.*)([;]+[ ]*)(secure)(.*)$" "$1$4"
Header edit Set-Cookie "^(.*)([;]+[ ]*)(SameSite=Strict|SameSite=Lax)(.*)$" "$1$4"
Header edit Set-Cookie "^(.*)([;]*)$" "$1; SameSite=None; Secure"
```

PHP

Na razini PHP aplikacija, mogu se ikoristiti sljedeći primjeri: https://github.com/GoogleChromeLabs/samesite-examples/blob/master/php.md

Od verzije PHP-a 7.3 moguće je definirati opciju u php.ini https://wiki.php.net/rfc/same-site-cookie

.NET

Na razini .NET aplikacija može se postaviti ponašanje u konfiguracijskoj datoteci web.config.

Prvi primjer:

```
<configuration>
 <system.web>
  <httpCookies sameSite="None" requireSSL="true" />
 <system.web>
<configuration>
```

Drugi primjer:

```
<configuration>
 <system.web>
  <sessionState cookieSameSite="None" /> <!-- No config attribute for Secure -->
  <httpCookies requireSSL="true" />
 <system.web>
<configuration>
```

Za više detalja i primjera proučite sljedeće poveznice:

https://docs.microsoft.com/en-us/aspnet/samesite/system-web-samesite

https://stackoverflow.com/a/57840284(link is external)

https://github.com/GoogleChromeLabs/samesite-examples/blob/master/aspnet.md

https://docs.microsoft.com/en-us/dotnet/api/system.web.httpcookie.samesite#System_Web_HttpCookie_SameSite

https://support.oneidentity.com/technical-documents/identity-manager/8.1/web-application-configuration-guide/10

SimpleSAMLphp

Od verzije SimpleSAMLphp 1.17.3 postoji opcija u datoteci config/config.php: "Add new options session.cookie.samesite and language.cookie.samesite that can be used to set a specific value for the cookies' SameSite attribute. The default it not to set it."

OIOSAML.Net

Od verzije OIOSAML.Net 2.0.3 omogućeno postavljanje atributa SameSite na vrijednost "None": https://www.digitaliser.dk/resource/5134883