Attacking Web Services

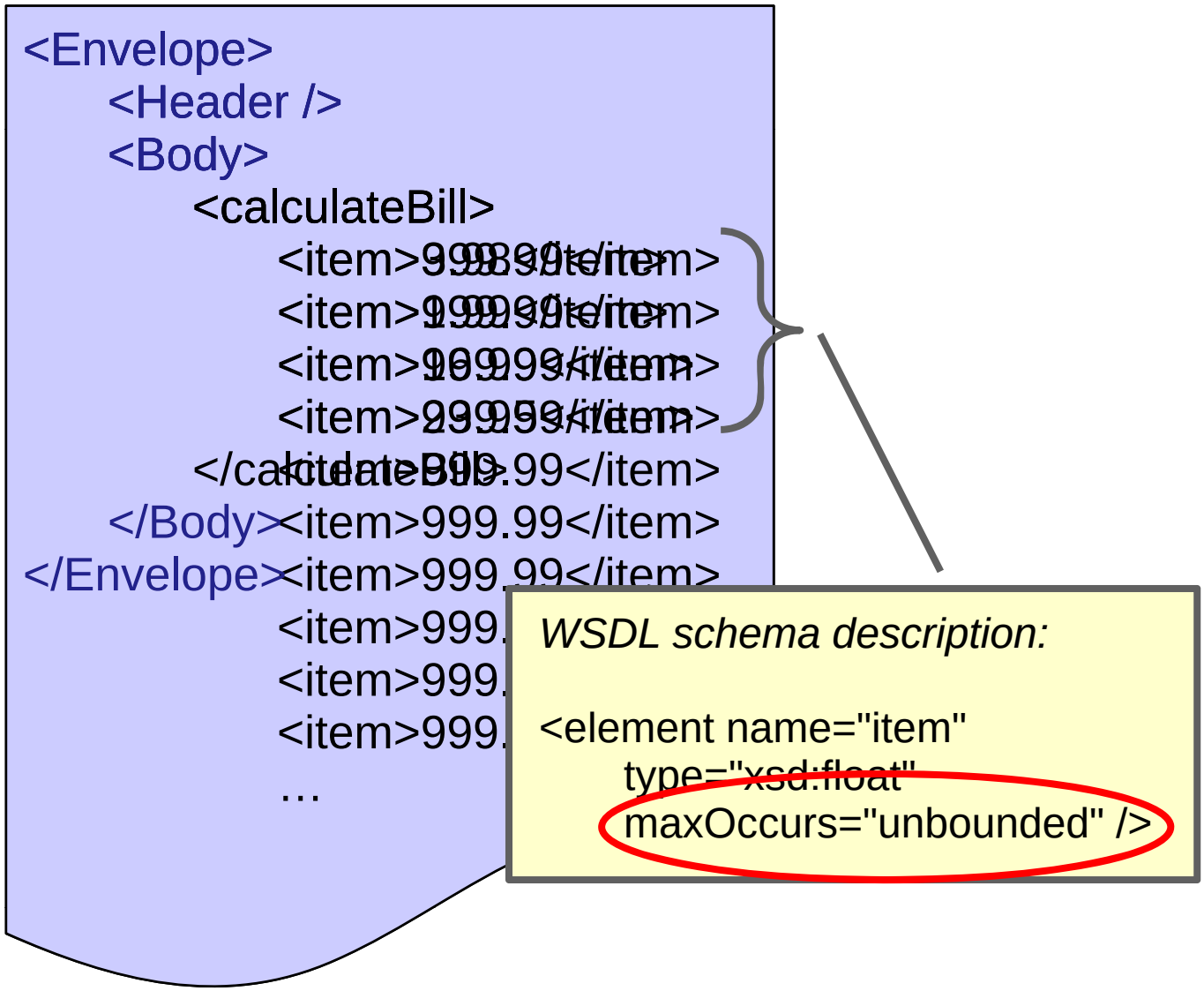
Overview

- Oversize Payload
- Coercive Parsing
- SOAPAction Spoofing
- Metadata Spoofing
- Attack Obfuscation
- WS-Addressing Spoofing
- BPEL State Deviation
- Signature Wrapping with Namespace Injection

Oversize Payload

Oversize Payload

Attack Concept:



Oversize Payload

Experiment Results:

Attack Name:	***Oversize Payload***
Attack Type:	Denial of Service
Target Framework:	Axis 1.4
Attack Message Size:	1.8 MB
Impact on Memory:	50 MB
Impact on CPU:	100 % for >1 min

Oversize Payload

Experiment Results:

Attack Name:	***Oversize Payload***
Attack Type:	Denial of Service
Target Framework:	Axis 1.4
Attack Message Size:	1.8 MB
Impact on Memory:	50 MB
Impact on CPU:	100 % for >1 min
Scale factor (Memory):	28

Coercive Parsing

Coercive Parsing

Attack Concept:

```
<Envelope>
    <Header />
    <Body>
        <visualize>
            <node>
                <node>
                    <leaf />
                    <leaf />
                </node>
                <node>
                    <leaf />
                </node>
            </node>
        </visualize>
    </Body>
</Envelope>
```

WSDL schema description:

```
<element name="node">
    <complexType>
        <choice>
            <element ref="node" />
            <element name="leaf" />
        </choice>
    </complexType>
</element>
```

Coercive Parsing

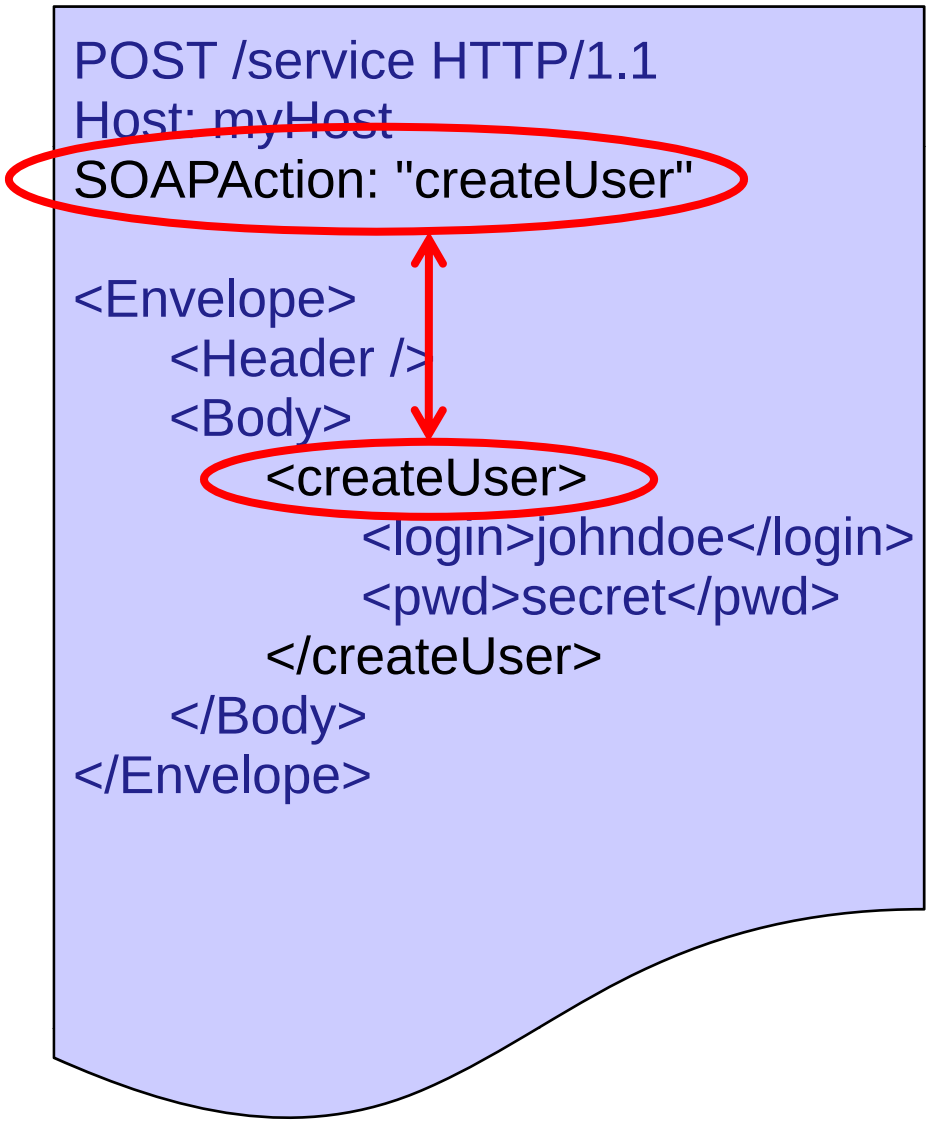
Experiment Results:

Attack Name:	***Coercive Parsing***
Target Framework:	Axis2
Number of Attack Messages:	1
Attack Message Size:	Endlessly continuable
Impact on CPU:	100% while the attack continued
Network transmission rate:	150 Byte per second

SOAPAction Spoofing

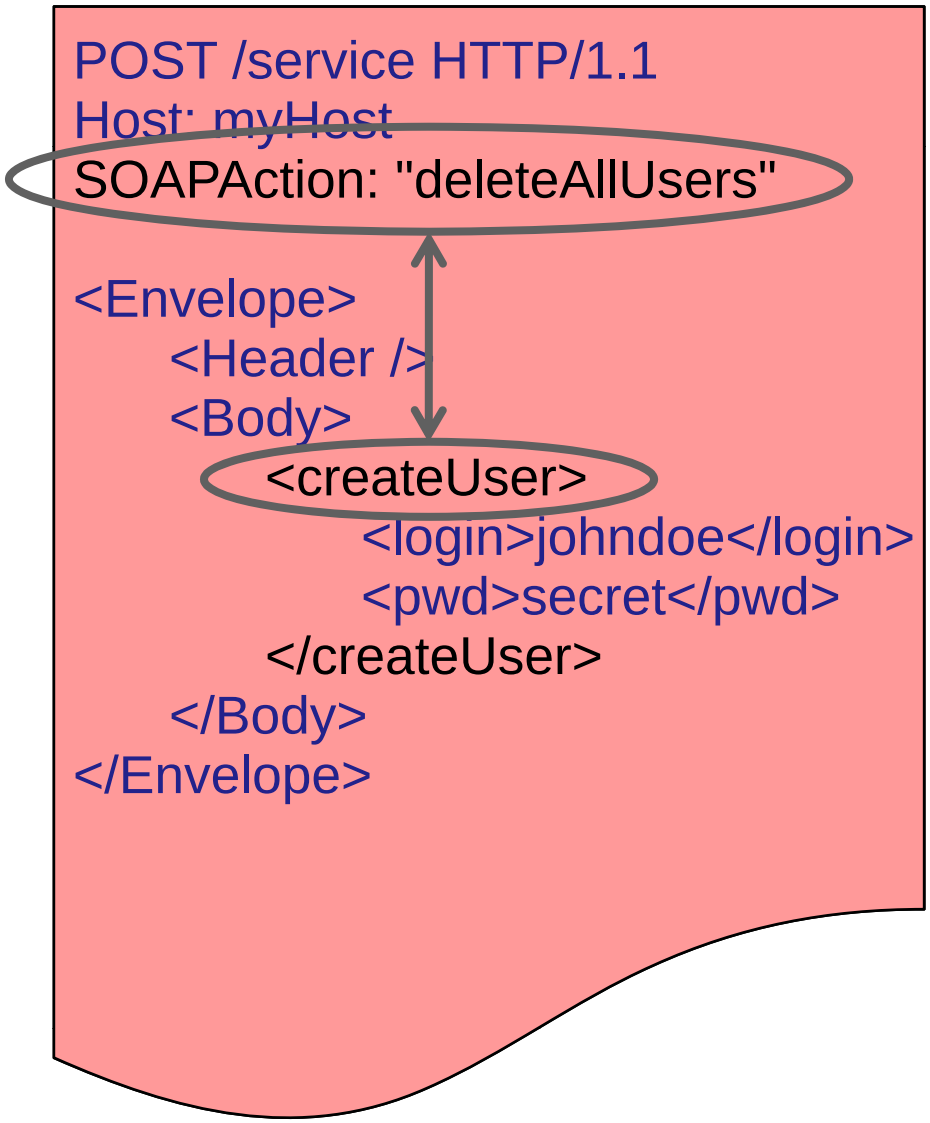
SOAPAction Spoofing

Attack Concept:



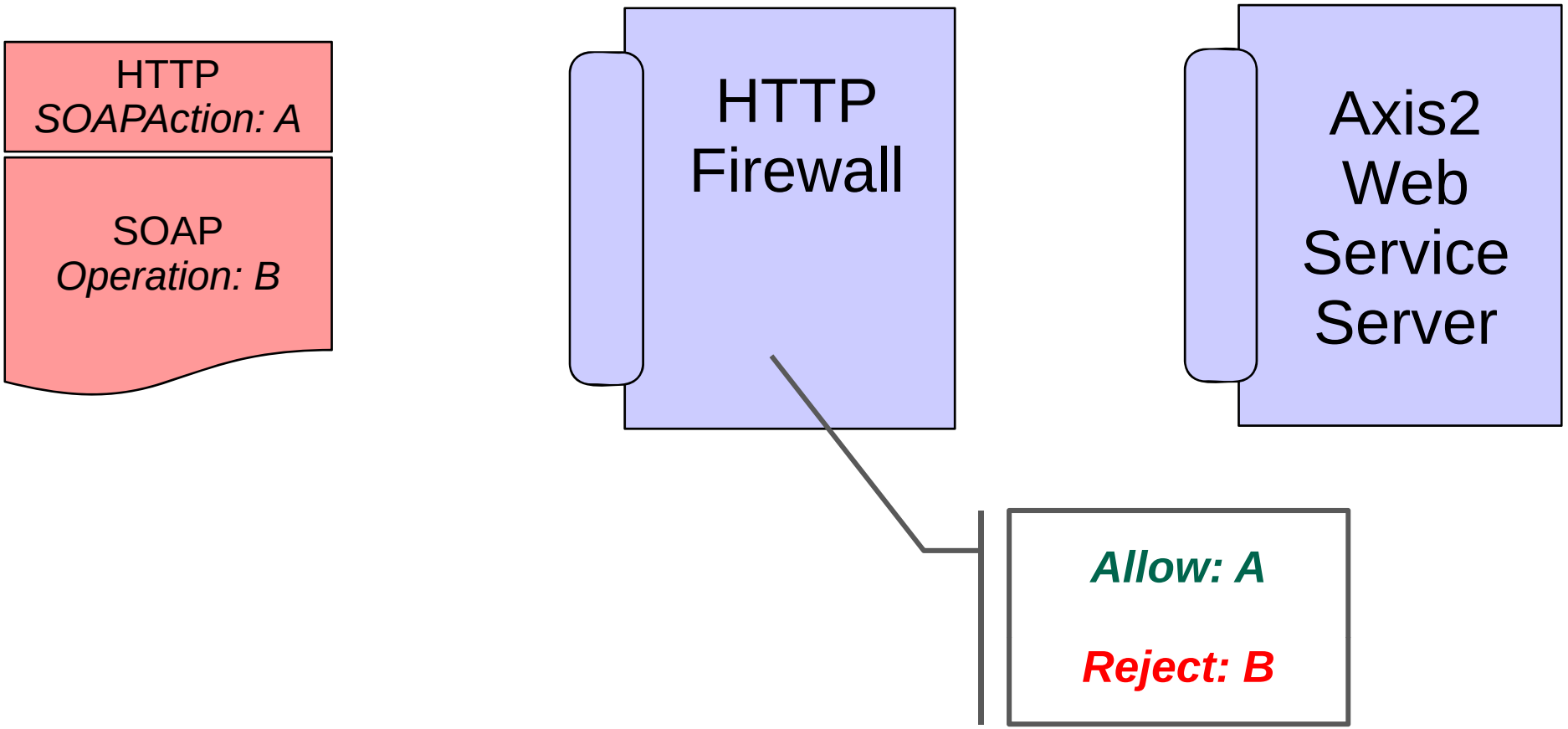
SOAPAction Spoofing

Attack Concept:



SOAPAction Spoofing

Axis2 impact:

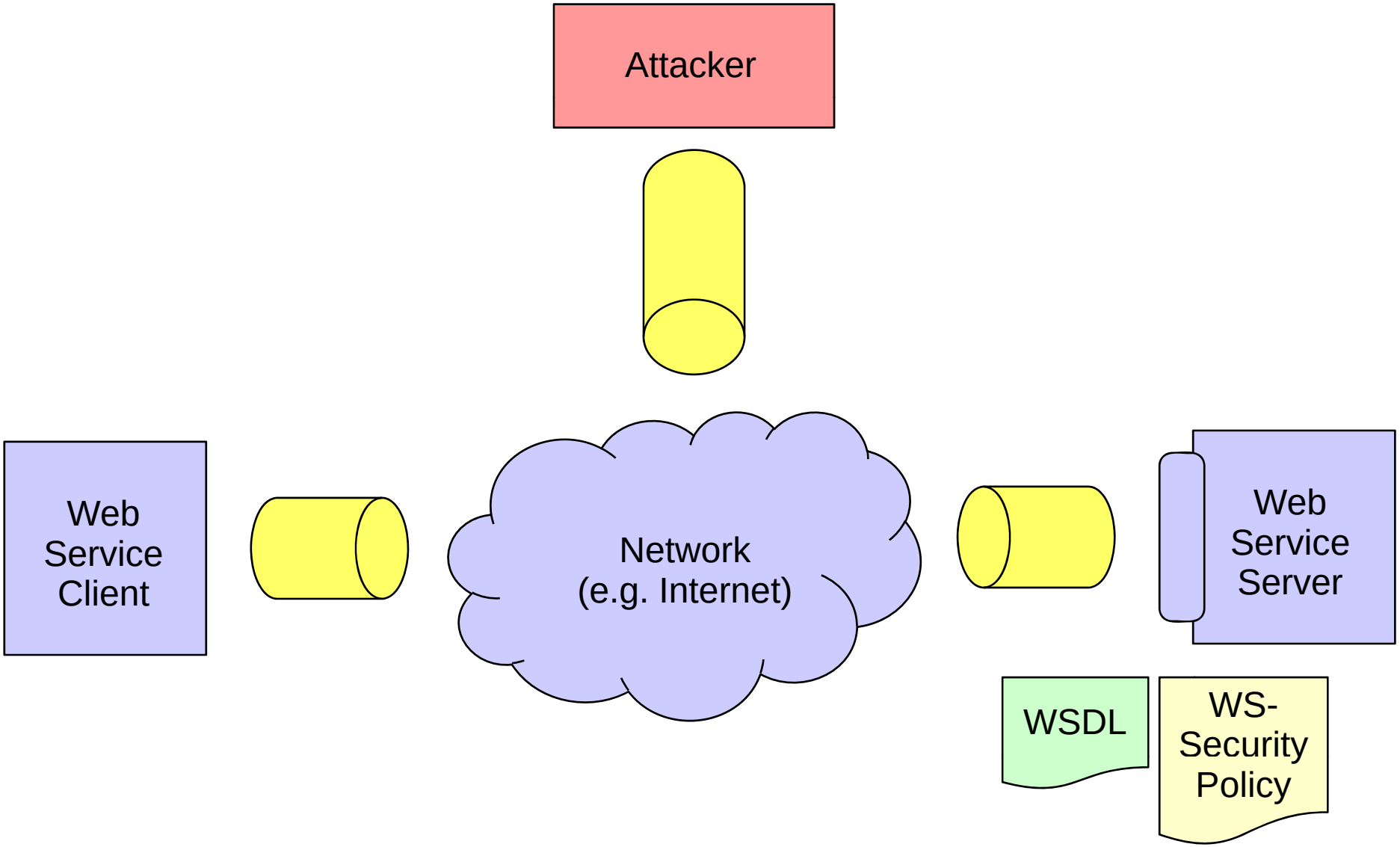


SOAPAction Spoofing

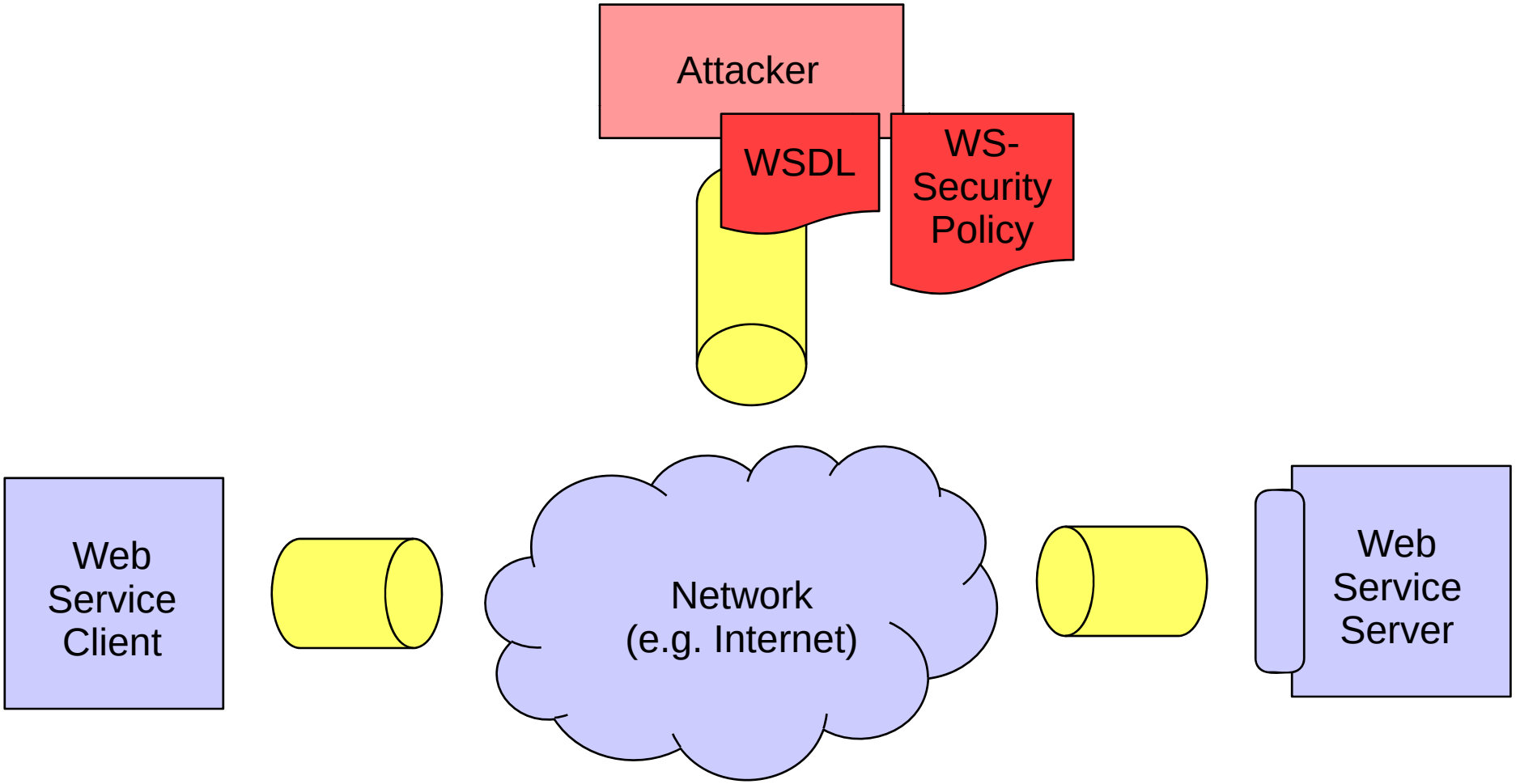
.NET impact:

Metadata Spoofing

Metadata Spoofing



Metadata Spoofing



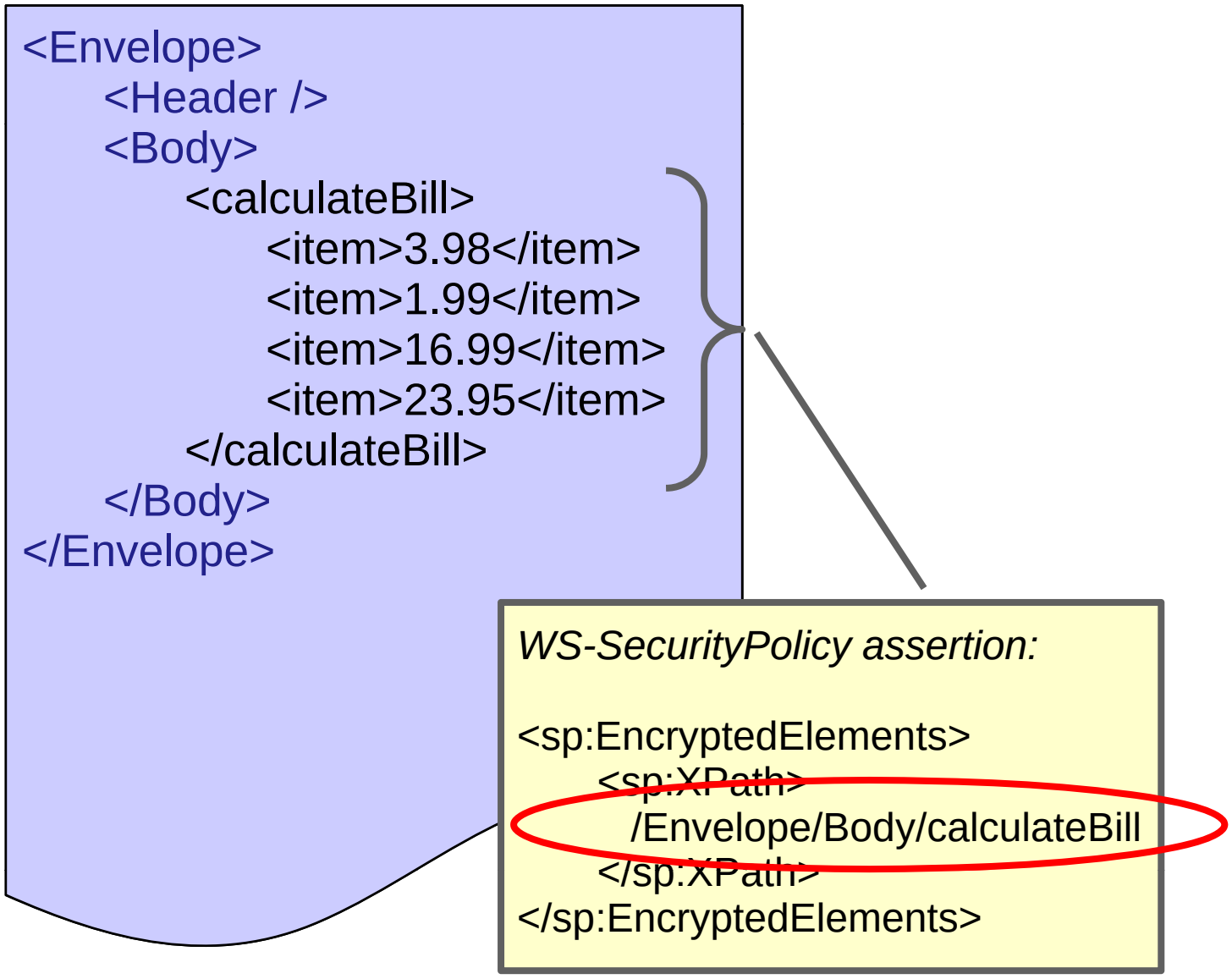
Metadata Spoofing

- Spoofed WSDL:
  - Change endpoint URL
    ➡ Man-in-the-middle scenario
  - Change message schema
    ➡ Add/remove/change/fake operations
  - Attach spoofed WS-SecurityPolicy
    ➡ Modify security assertions
- Spoofed WS-SecurityPolicy:
  - Change cryptographic algorithms to use
    ➡ Encryption becomes breakable
  - Remove security assertions
    ➡ Eavesdropping and data modification

Attack Obfuscation

Attack Obfuscation

Attack Concept:



Attack Obfuscation

Attack Concept:

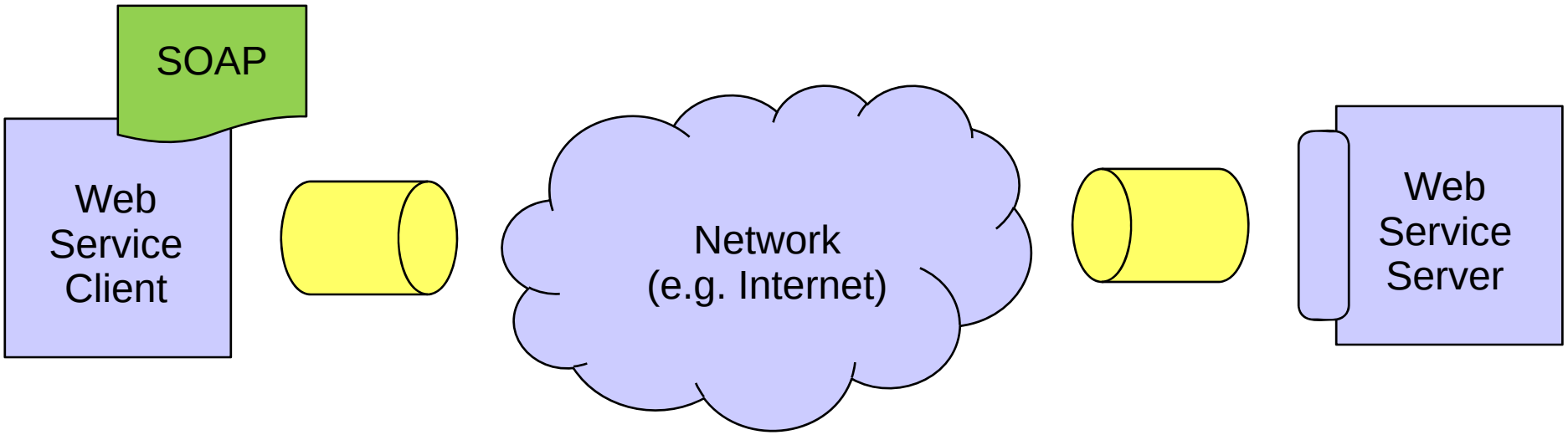
Attack Obfuscation

Experiment Results:

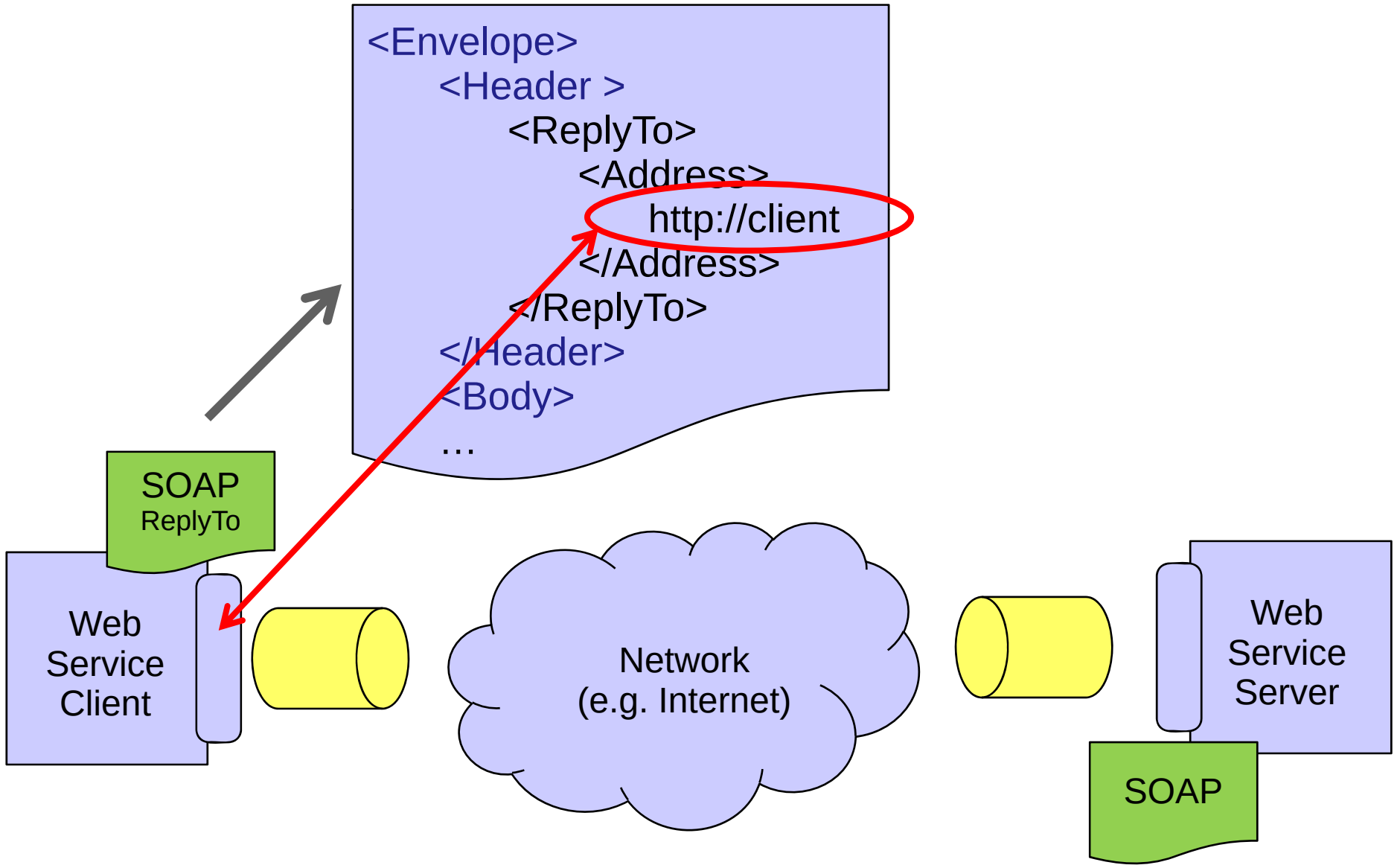
Attack Name:	***Attack Obfuscation***
Attack Type:	Denial of Service
Target Framework:	Rampart 1.0 + Axis2
Attack Message Size:	1 MB
Impact on Memory:	90 MB
Impact on CPU:	100 % for 23 sec
Scale factor (Memory):	90

WS-Addressing Spoofing

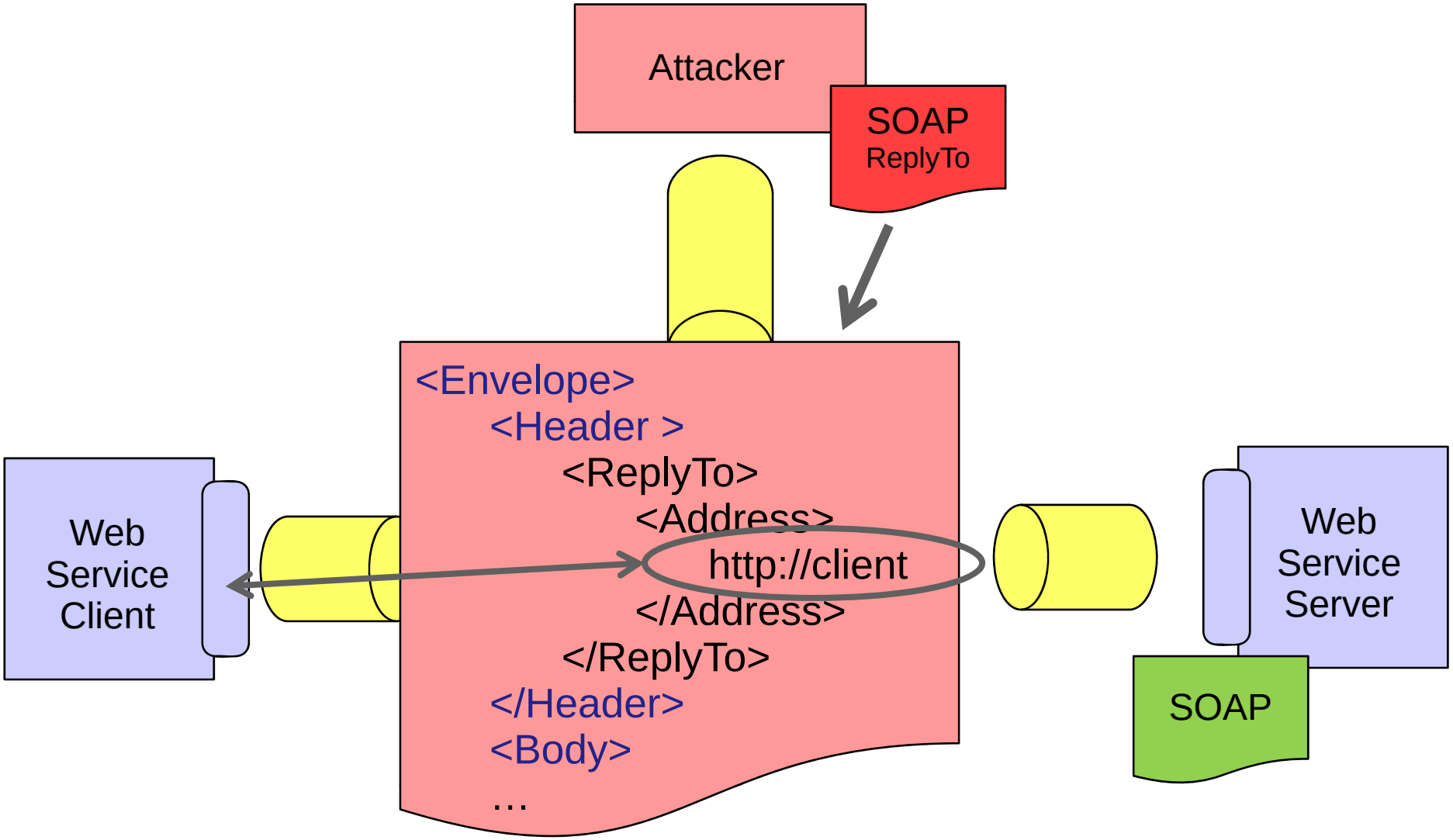
WS-Addressing Spoofing



WS-Addressing Spoofing



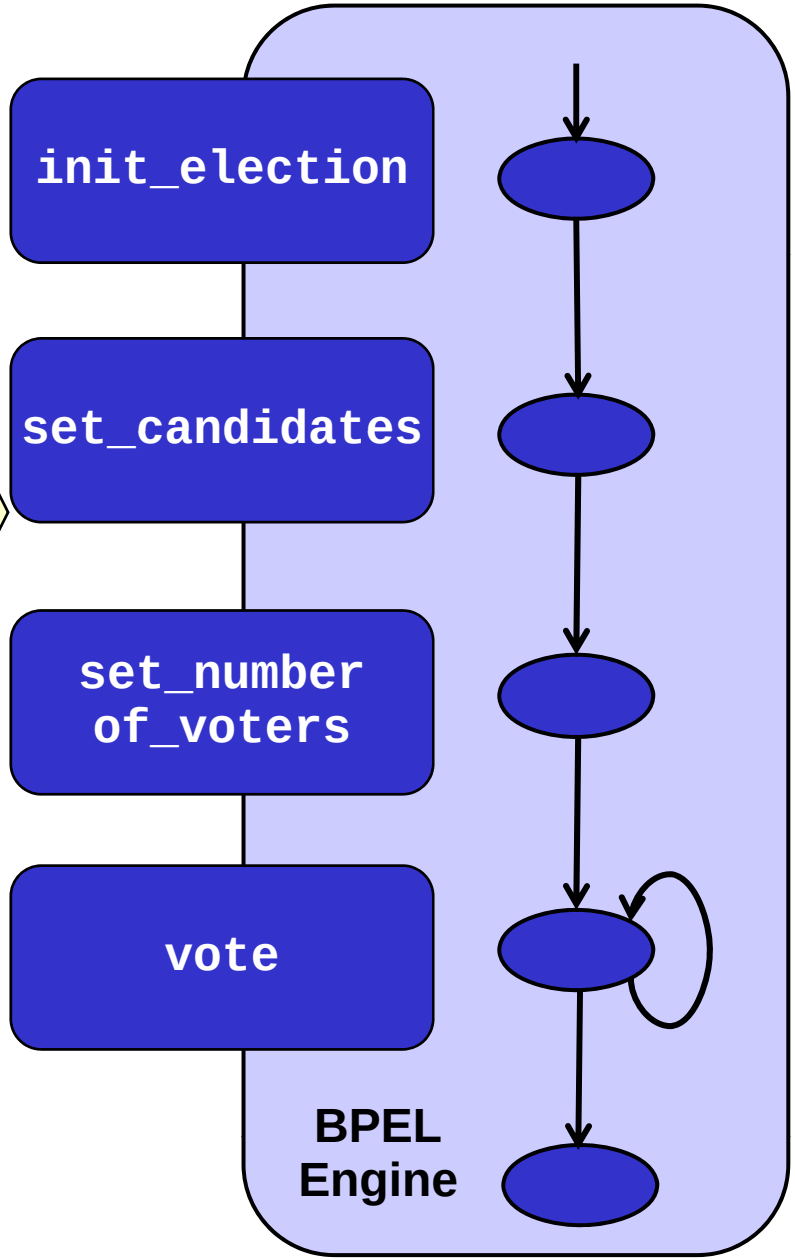
WS-Addressing Spoofing



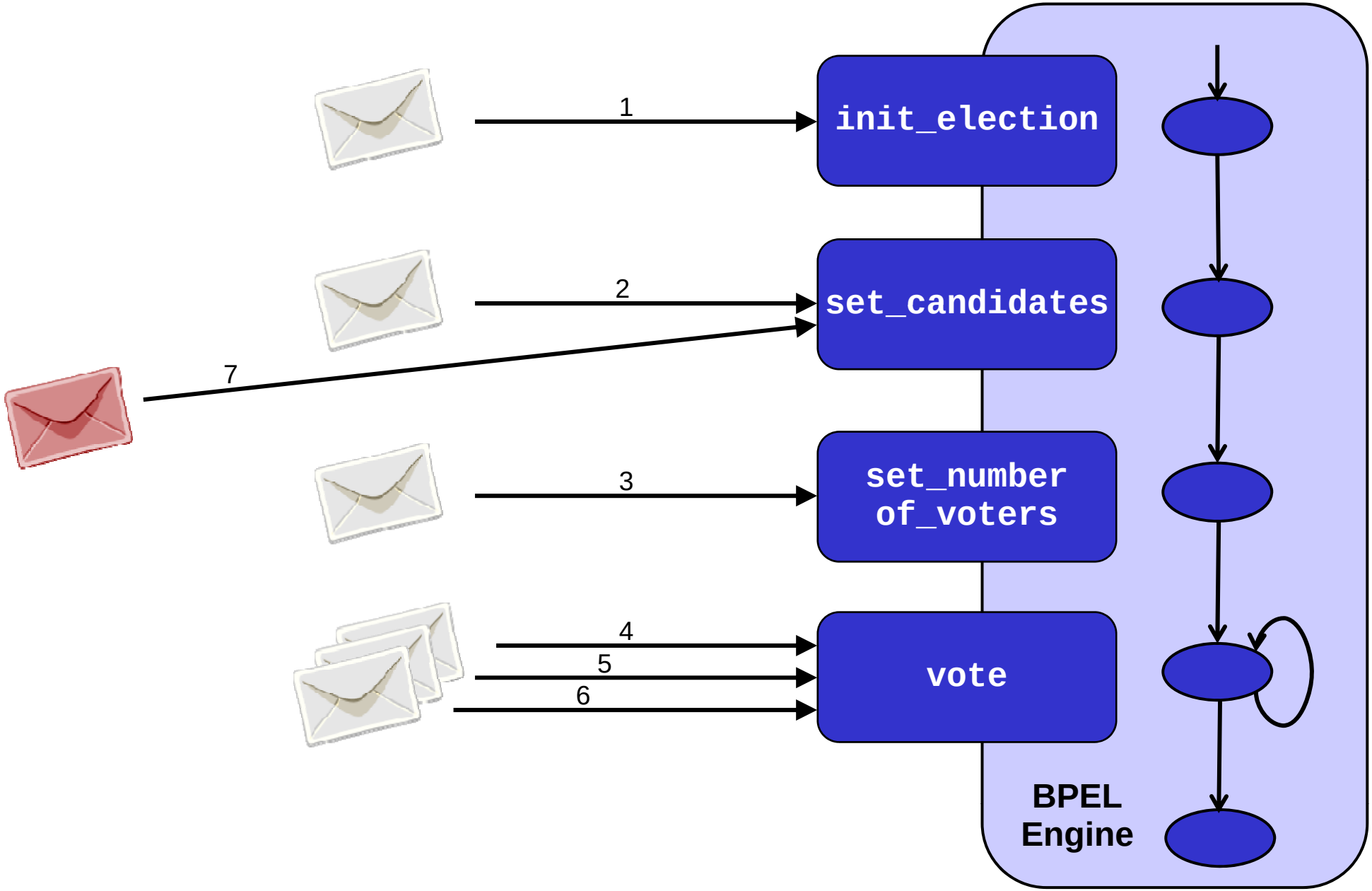
BPEL State Deviation

BPEL State Deviation

```
<process>
    <sequence>
        <receive operation="init_election" />
        <receive operation="set_candidates" />
        <receive operation="set_number_of_voters" />
        <while condition="voting_not_complete()">
            <receive operation="vote" />
        </while>
        <invoke operation="announce_winner" />
    </sequence>
</process>
```



BPEL State Deviation



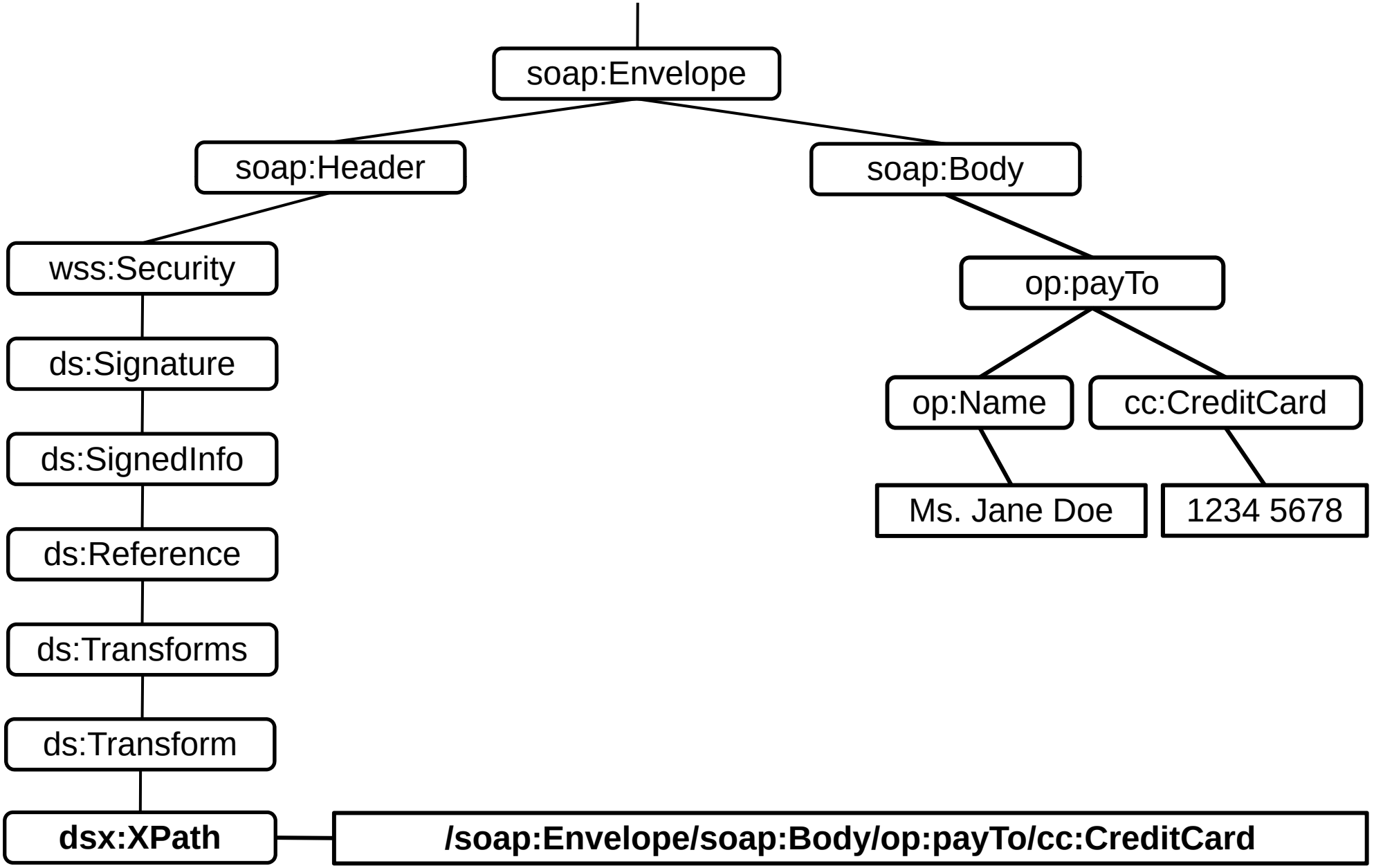
BPEL State Deviation

Experiment Results:

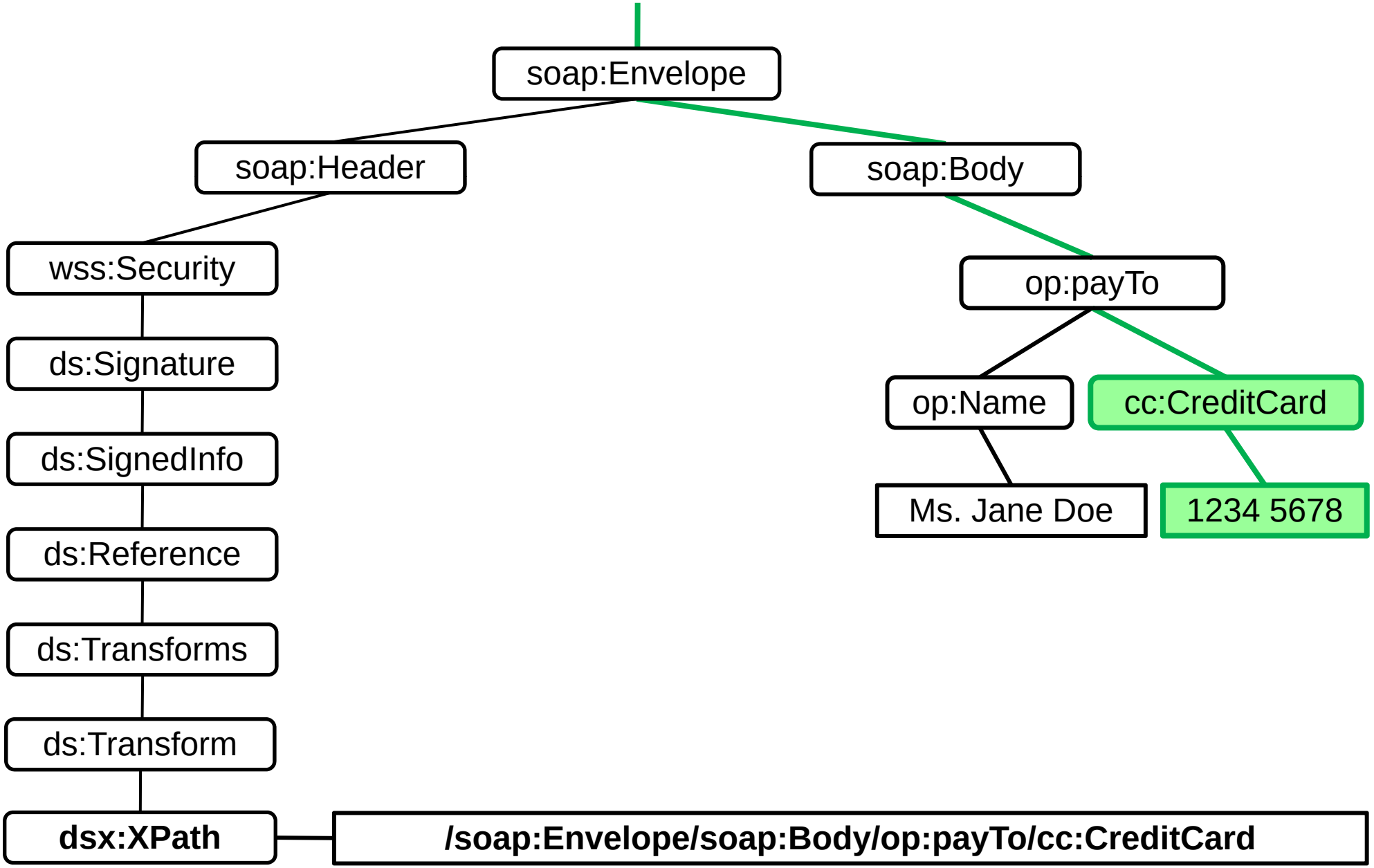
Attack Name:	***BPEL State Deviation***
Attack Type:	Denial of Service
Target Framework:	Oracle BPEL Process Manager 10.1
Attack Message Size:	1000 × 500 Byte = 0.5 MB
Impact on Memory:	350 MB
Impact on CPU:	100 % for 2 hours
Scale Factor (Memory):	700

Signature Wrapping with Namespace Injection

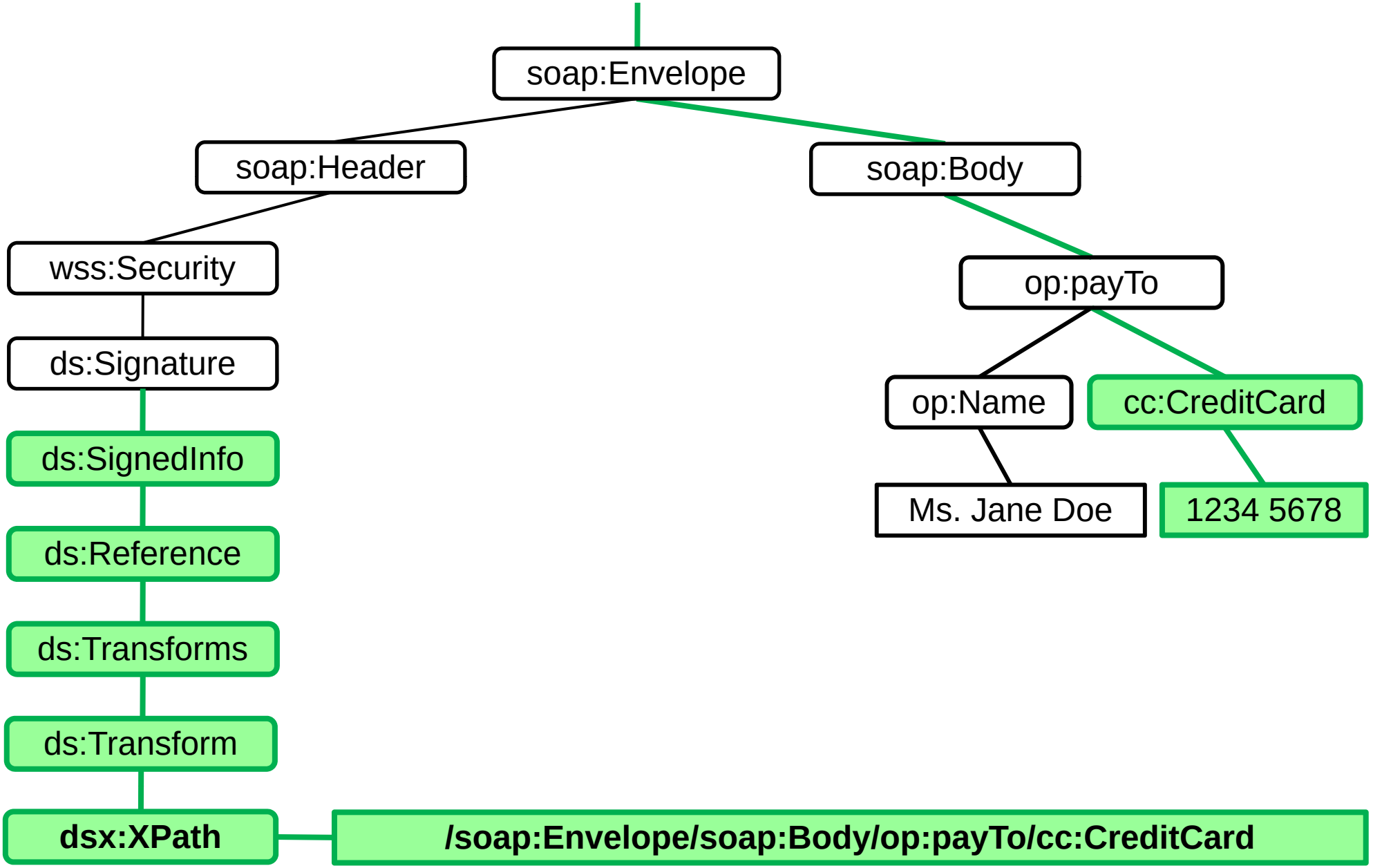
Signature Wrapping with Namespace Injection



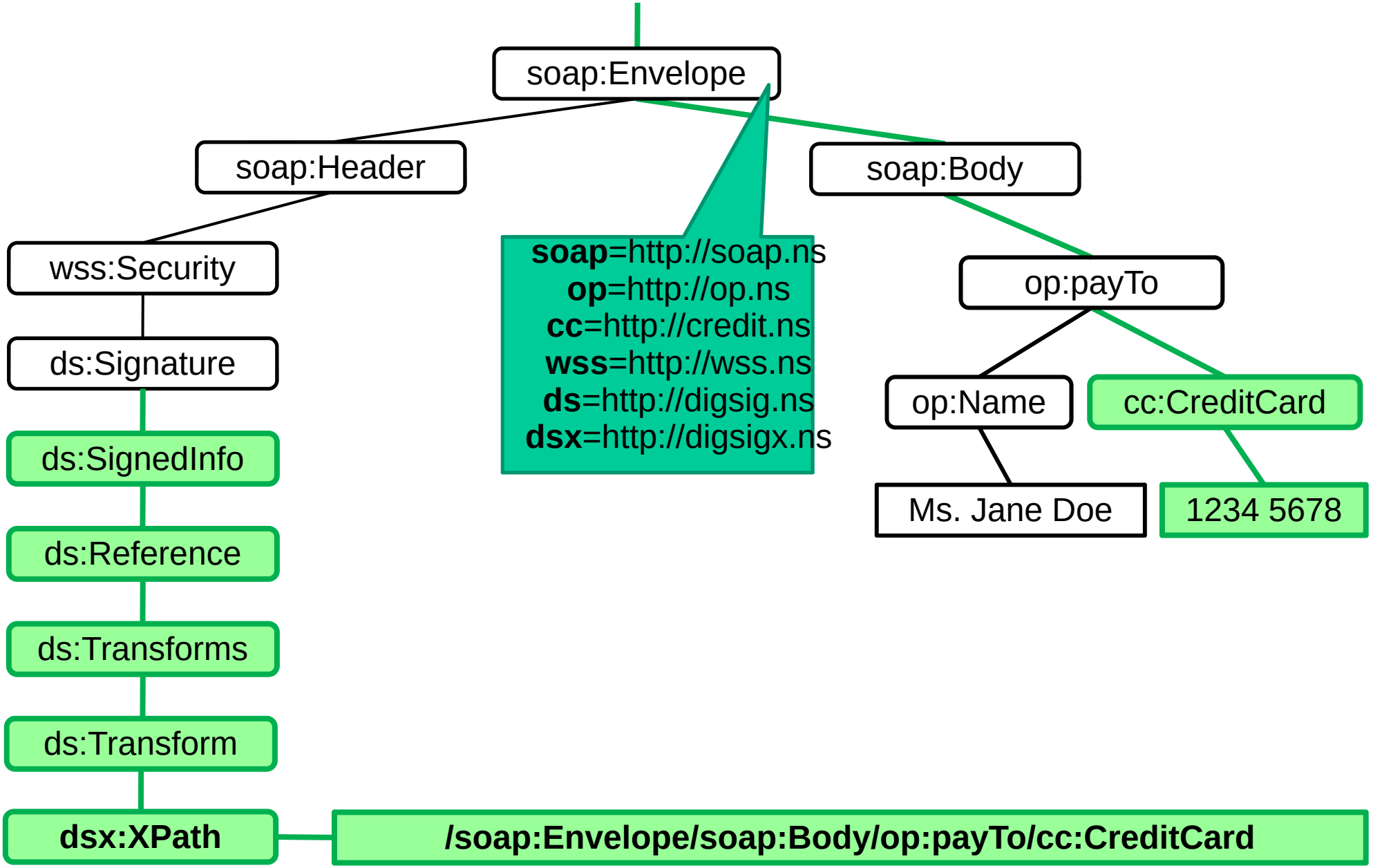
Signature Wrapping with Namespace Injection



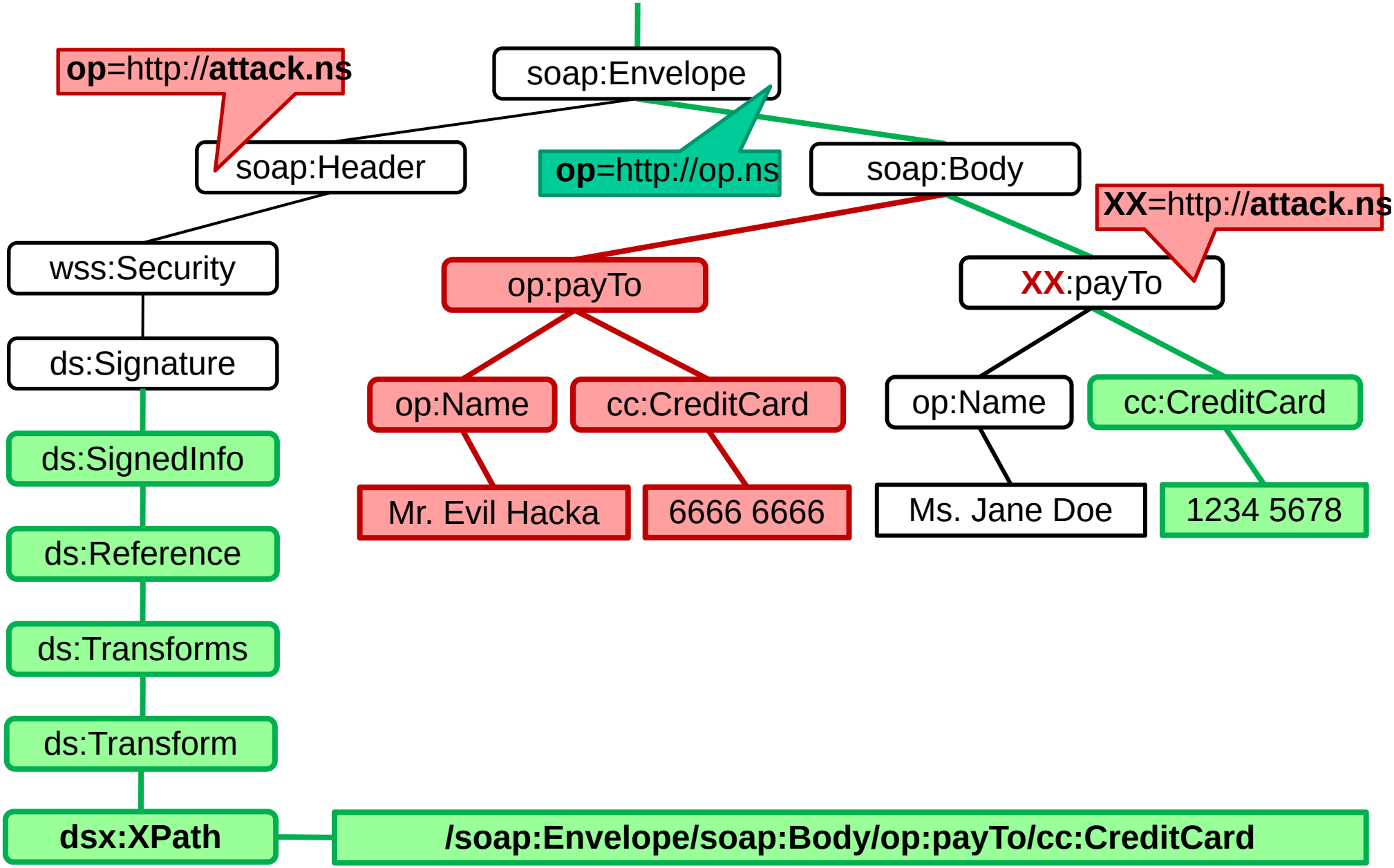
Signature Wrapping with Namespace Injection



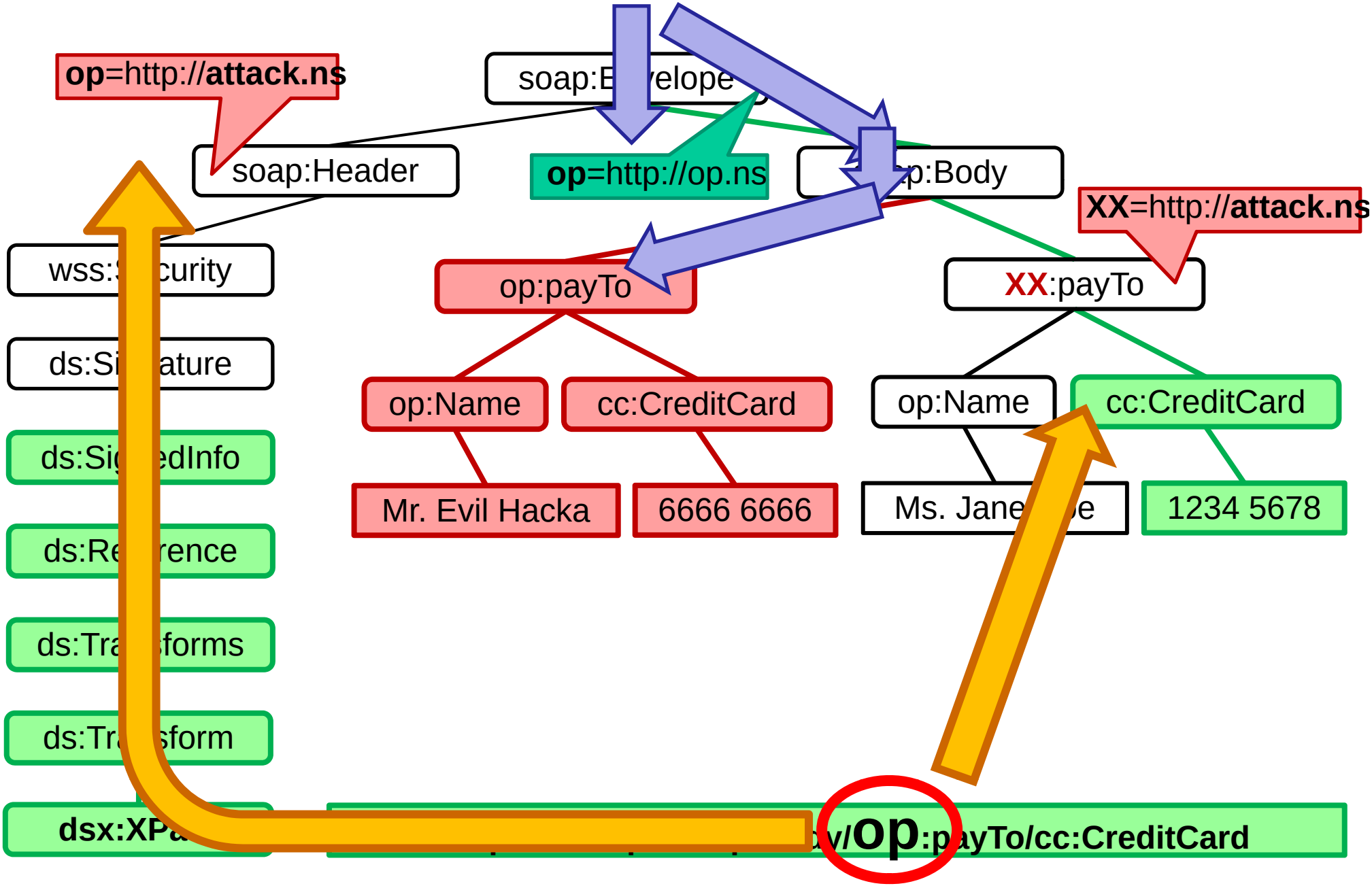
Signature Wrapping with Namespace Injection



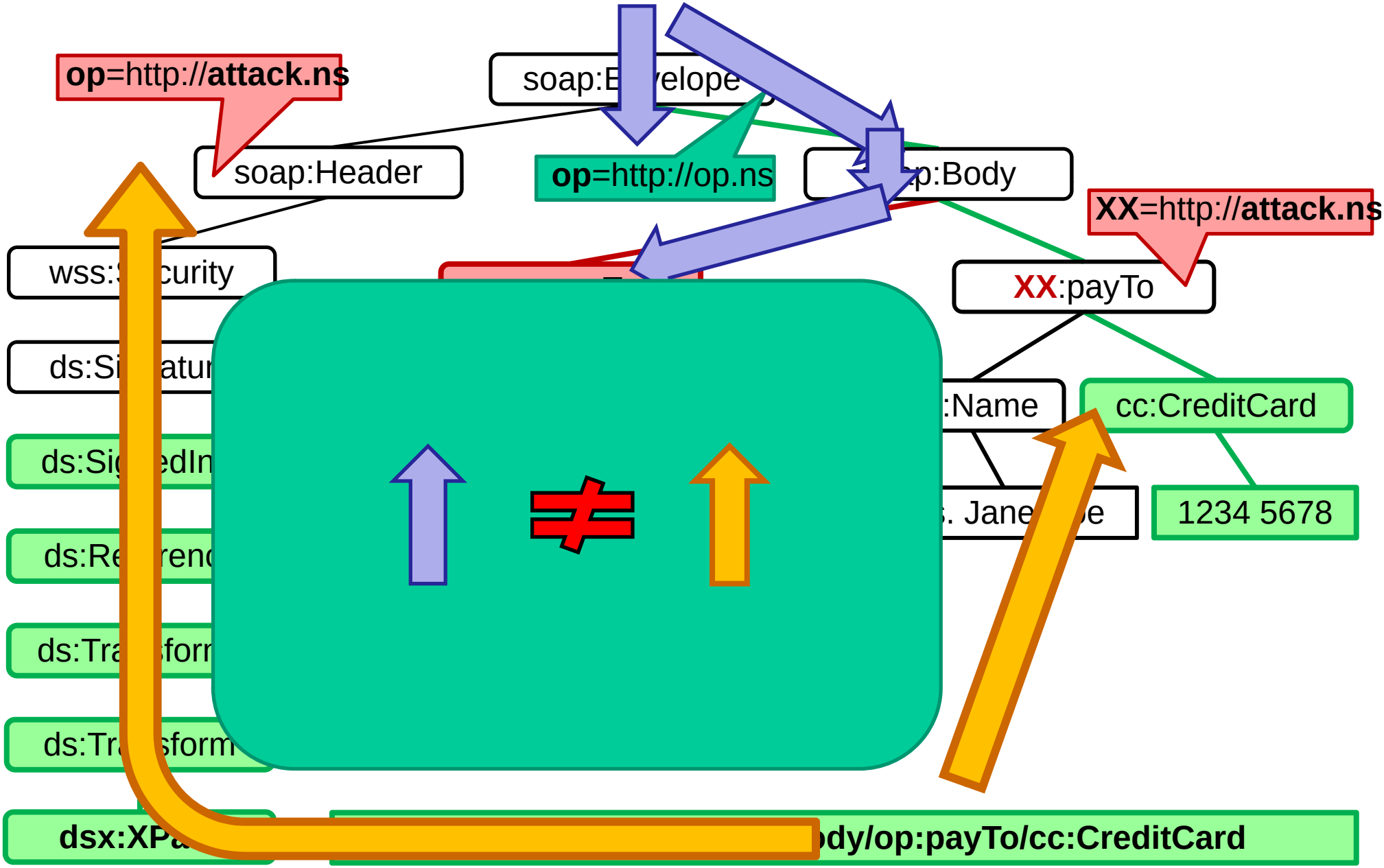
Signature Wrapping with Namespace Injection



Signature Wrapping with Namespace Injection



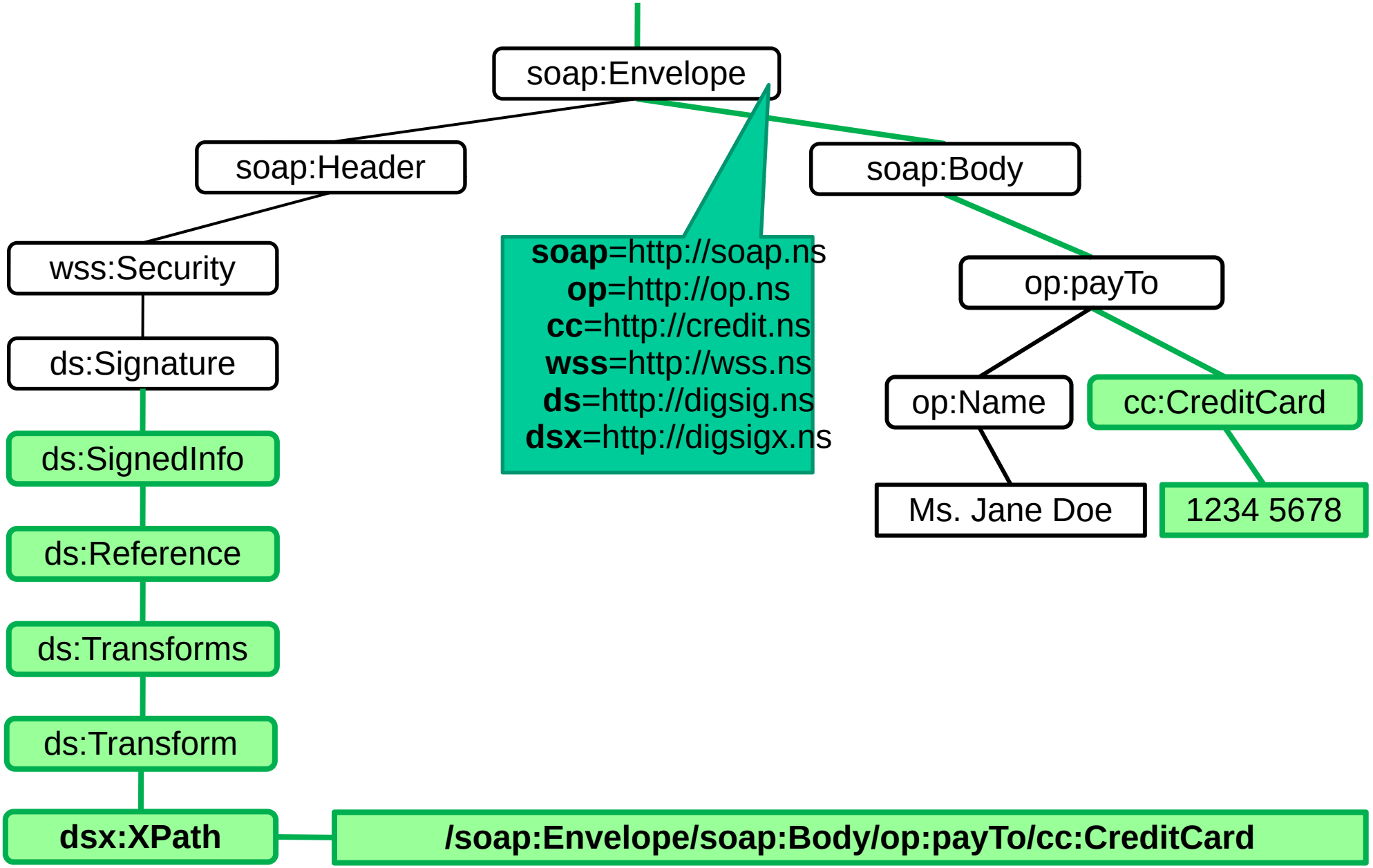
Signature Wrapping with Namespace Injection



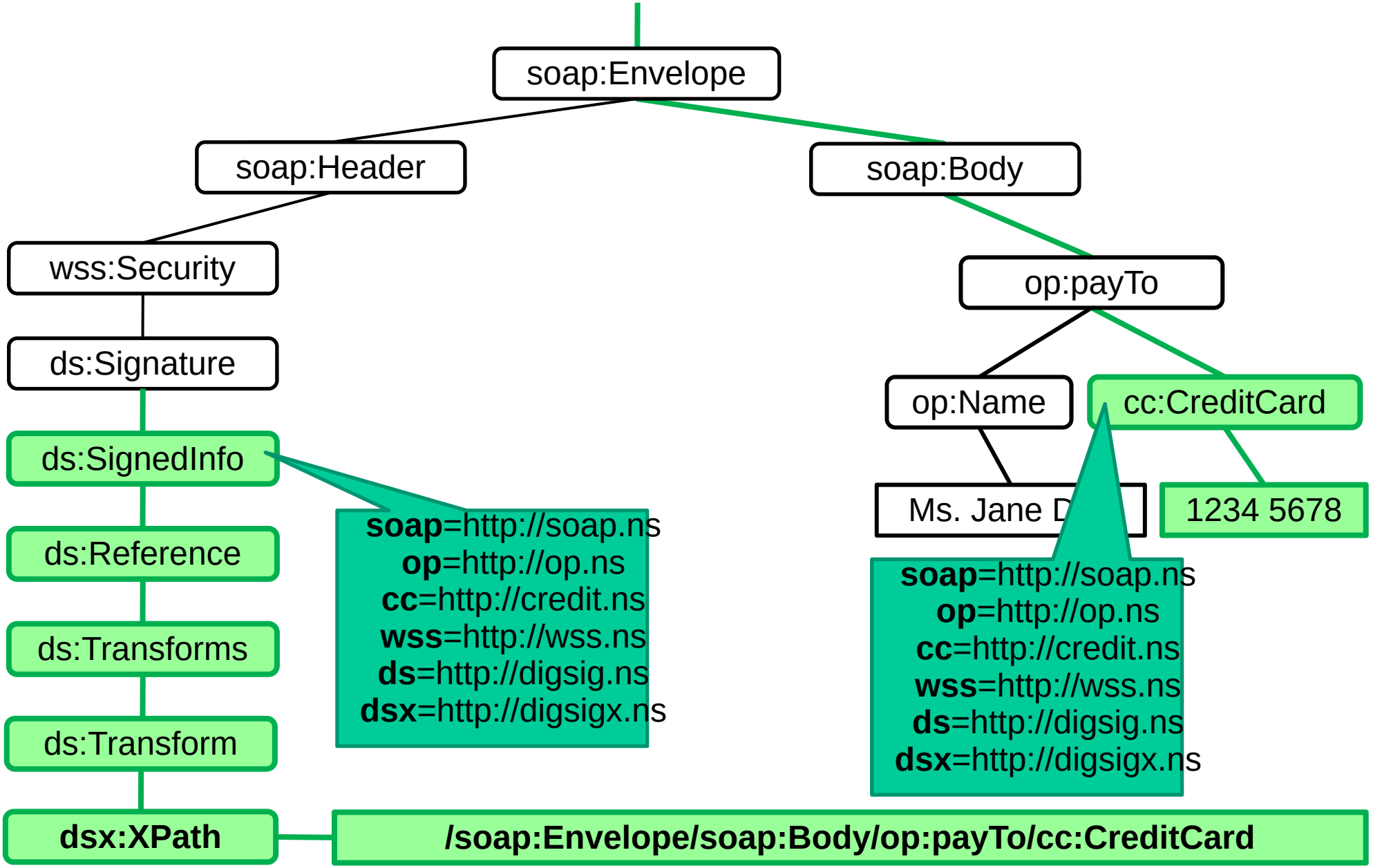
Signature Wrapping with Namespace Injection

By mapping the
same namespace prefix
to
different namespace urls
at certain positions
within an XML document,
an attacker can „inject“ contents
that are processed
as if they were signed.

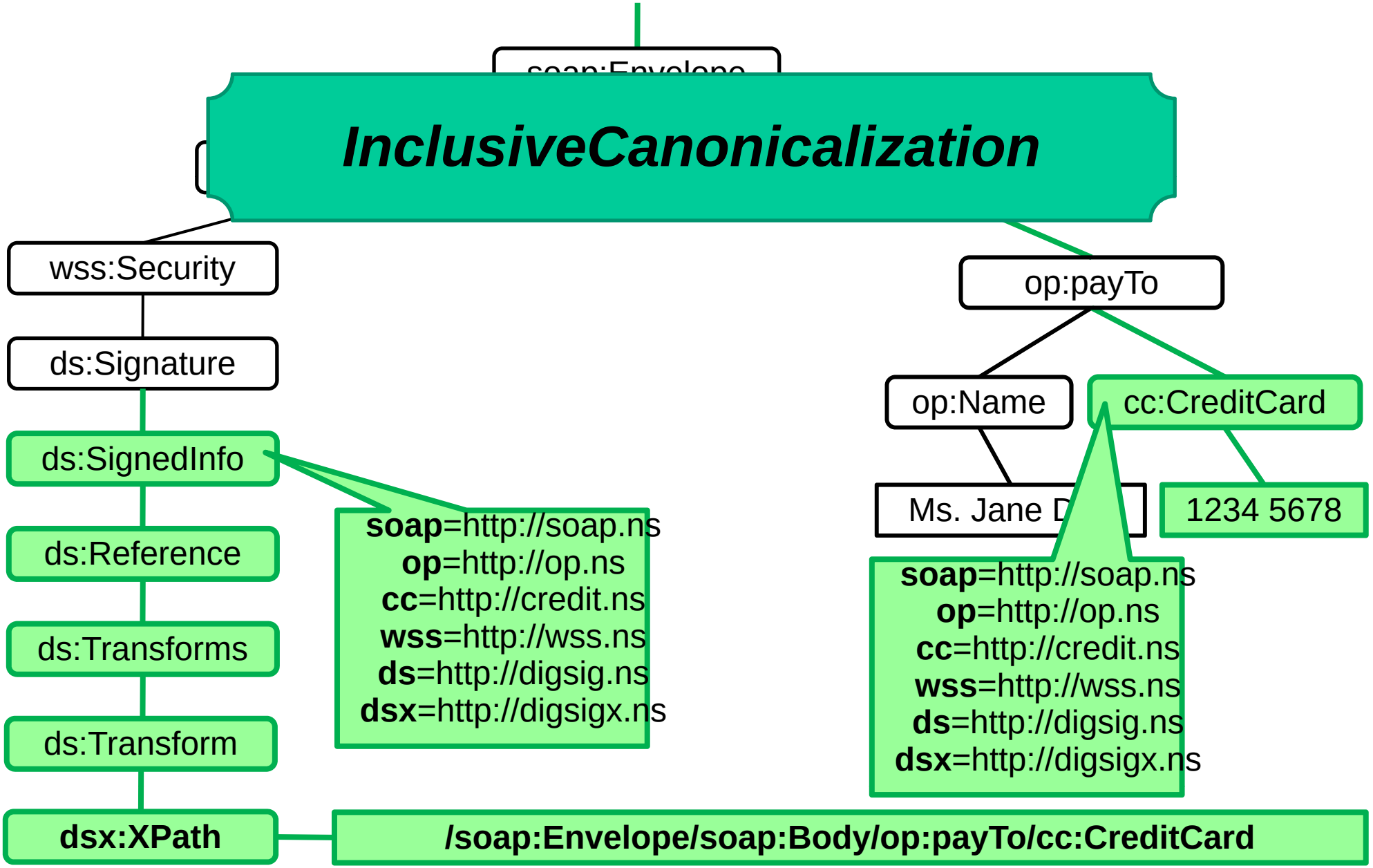
Signature Wrapping with Namespace Injection



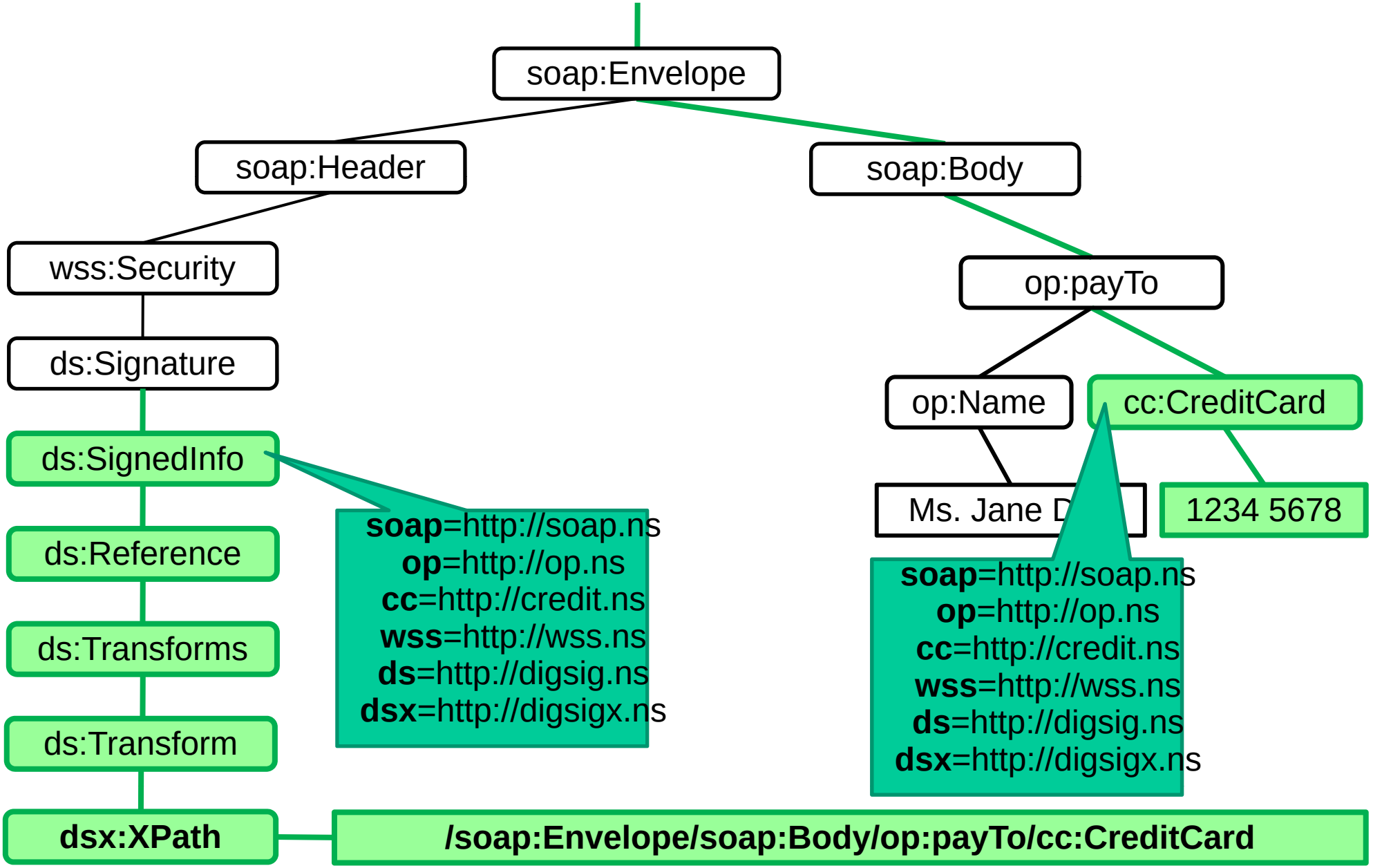
Signature Wrapping with Namespace Injection



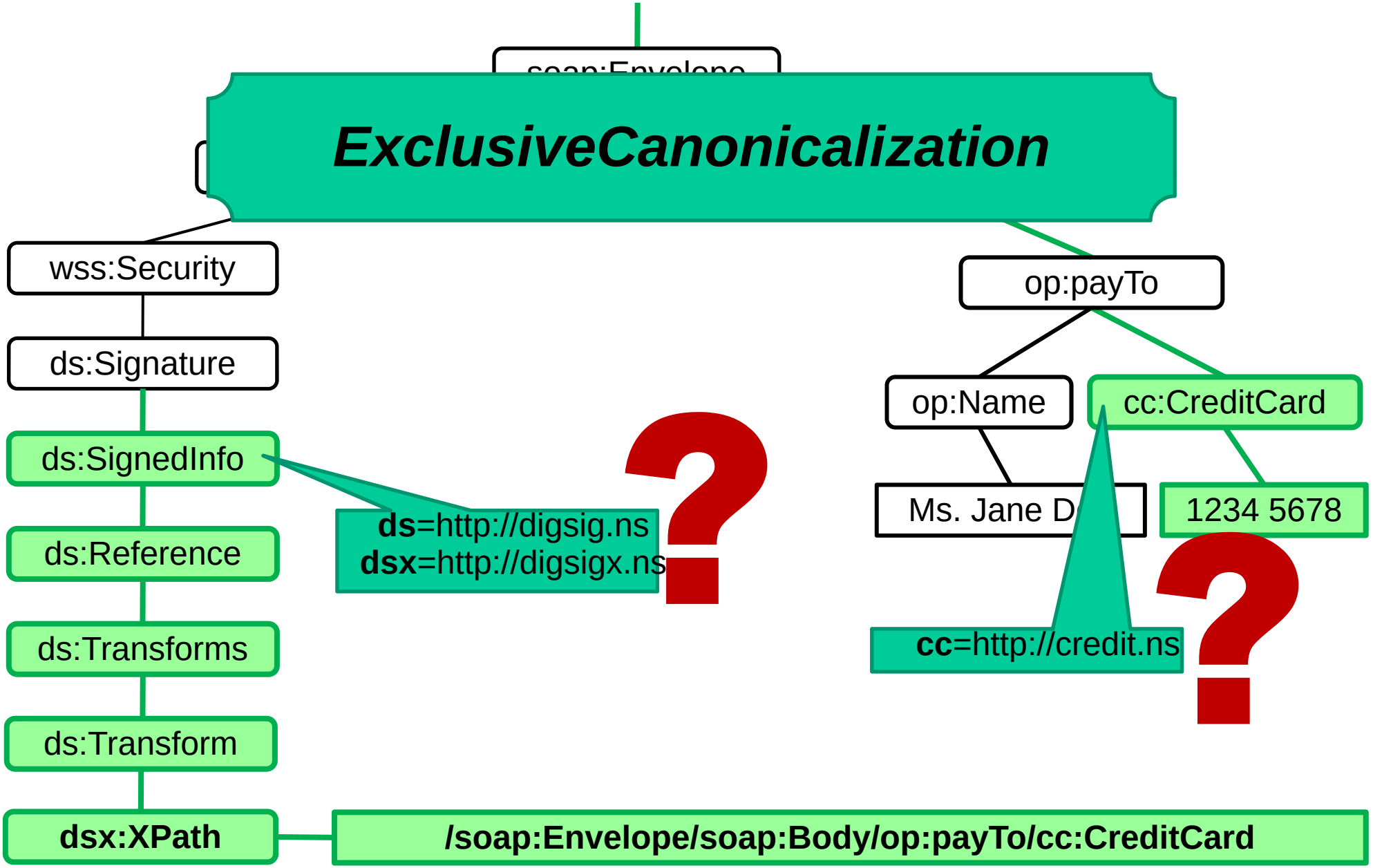
Signature Wrapping with Namespace Injection



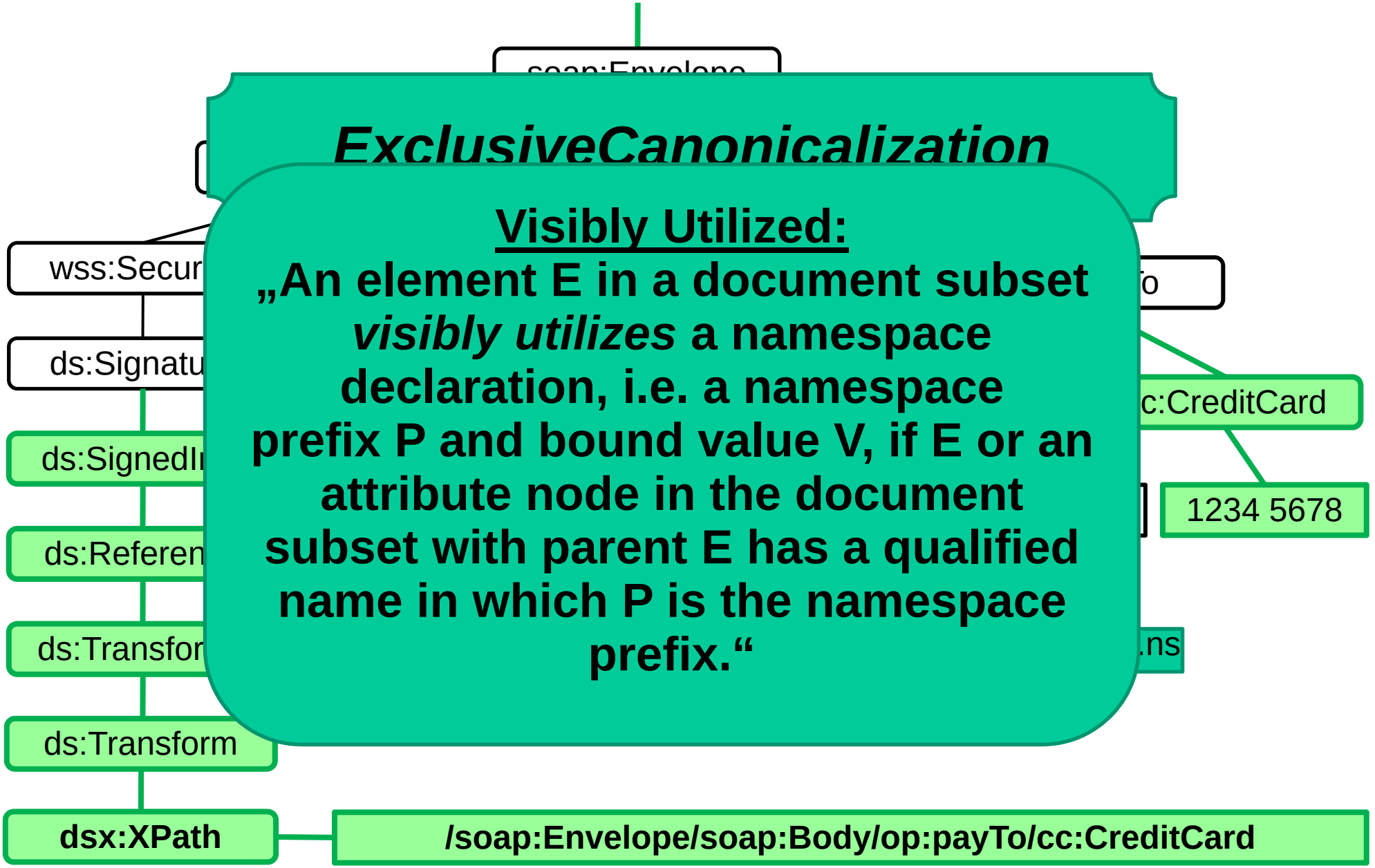
Signature Wrapping with Namespace Injection



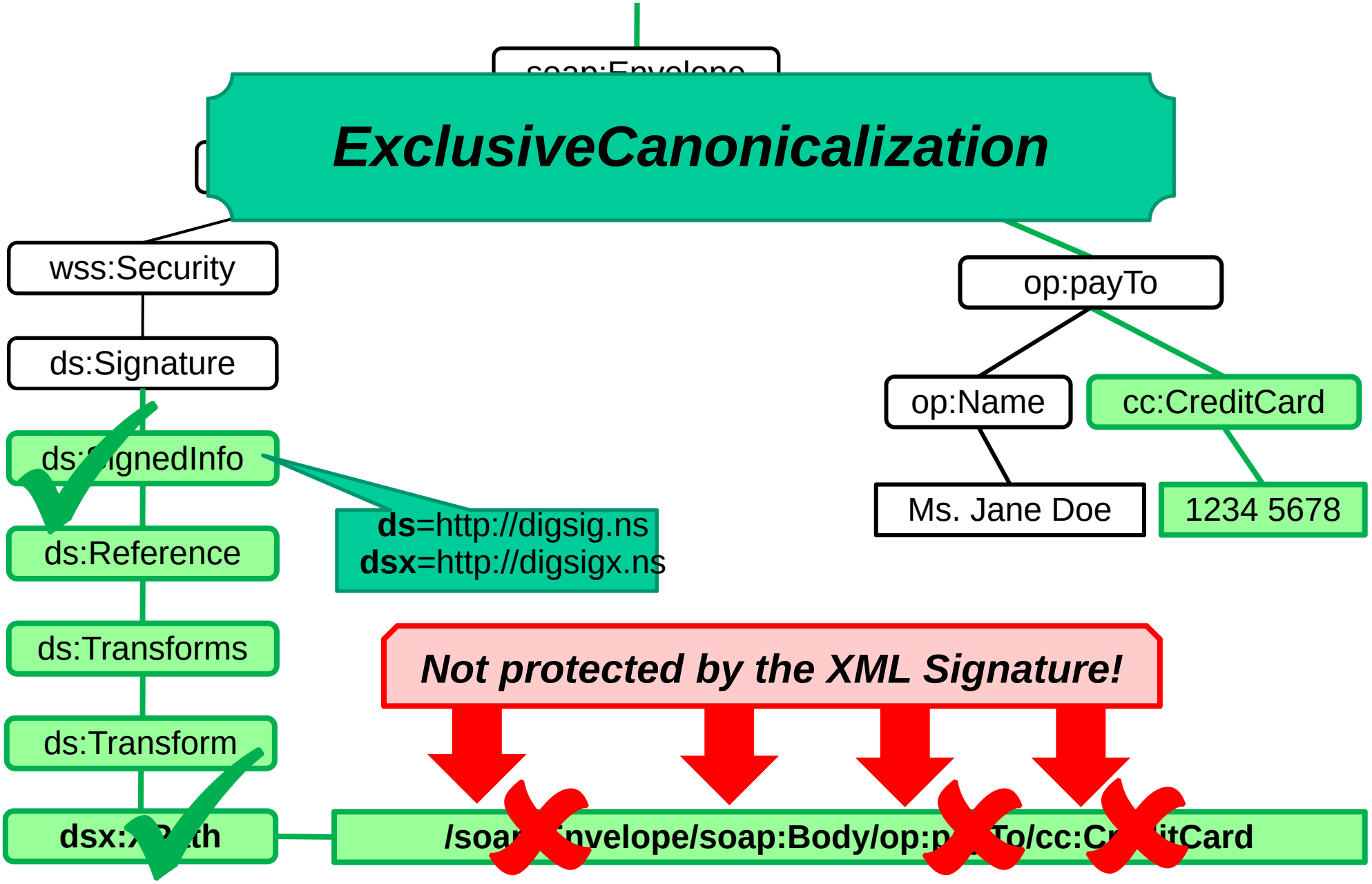
Signature Wrapping with Namespace Injection



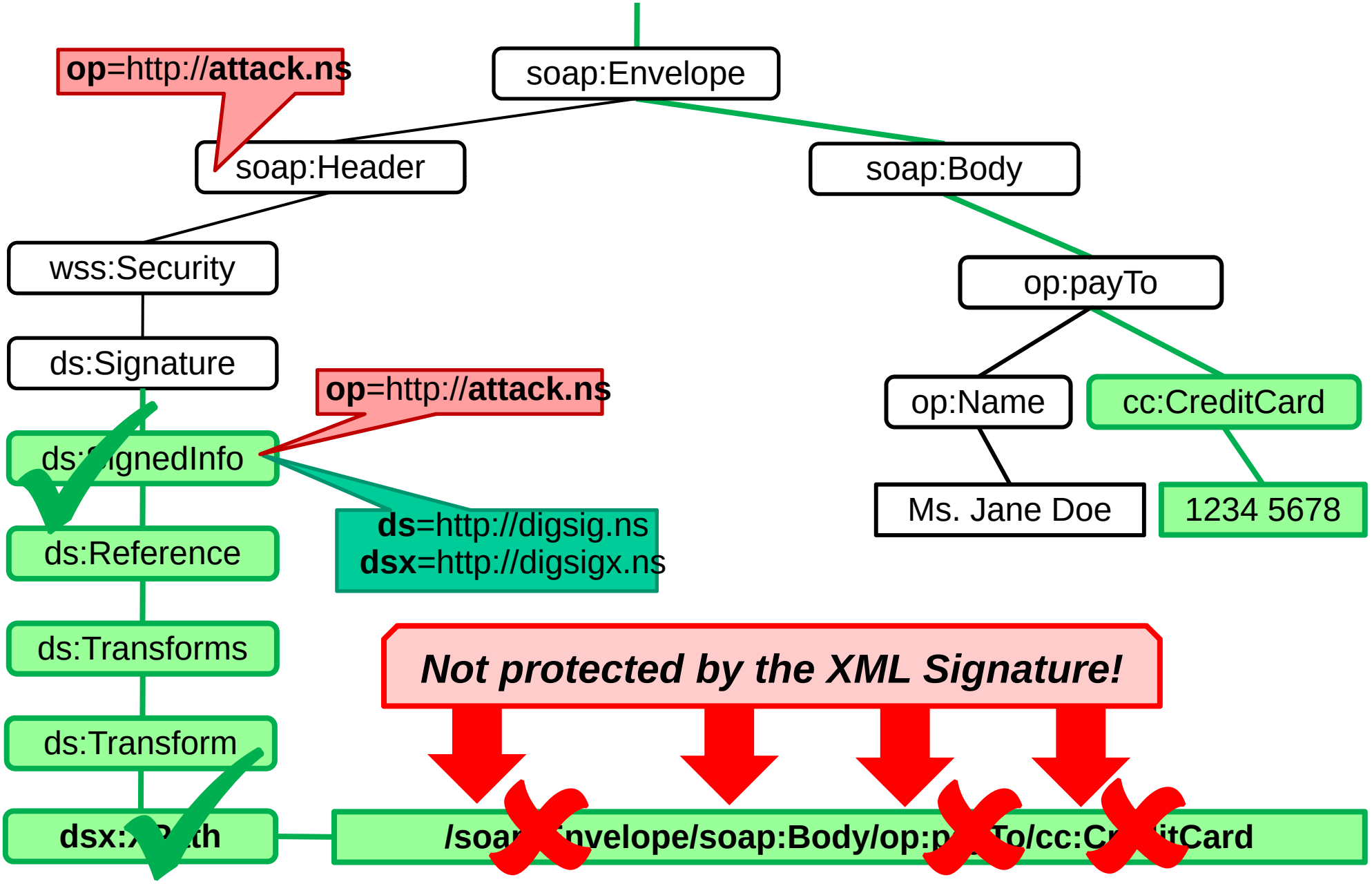
Signature Wrapping with Namespace Injection



Signature Wrapping with Namespace Injection



Signature Wrapping with Namespace Injection



Schöne Semesterferien!