Flooding Attack Issues of Web Services and Service-Oriented Architectures

Meiko Jensen
Department for Computer Science, University of Kiel, Germany
mje@informatik.uni-kiel.de

Nils Gruschka*
NEC Laboratories Europe, IT Research Division, St. Augustin, Germany
gruschka@it.neclab.eu

Abstract: The service-oriented architecture paradigm slowly matures towards some kind of "Web Service Internet" where basically everybody may use the services others provide. Though this evolution enables lots of opportunities for electronic business, it also induces many new security issues to consider. One important security threat to SOA consists in request floodings, which—being intentional or accidental—may rapidly lead to Denial-of-Service and other kinds of malfunctions.

In this paper, we will reconsider some of the known flooding attacks on Web Services, advance to flooding issues of basic service compositions, and finally derive some conclusions for security considerations of service-oriented architectures in general.

1 Introduction

In April/May 2007, the Estonian governmental and commercial communication infrastructure suffered from a massive Denial-of-Service attack performed via the Internet [Smi07]. Though retrospective analyses uncovered it to be merely an "angry mob" rather than a planned act of cyber-terrorism or Internet warfare [Bre07], its impact on Estonia's economy and communication infrastructure was tremendous.

Investigating the attack techniques used, it is valid to state that the particular attacks did not employ any new or unknown vulnerabilities or attack patterns. According to [Naz07], most of the attacks employed ICMP or TCP flooding techniques or regular web request floodings. The high severity of the achieved impact is due to the fact that the attackers employed botnets and similar techniques for performing the attacks.

In this paper, we will discuss some of the next big advances in network-based flooding attack techniques. We will show some examples on how threatening these new attack techniques (targeting Web Services) are today, and we will derive implications on what kinds of security problems might arise in a future "Web Service Internet"

[*] The work was performed while Nils Gruschka was a Ph.D. student at the University of Kiel

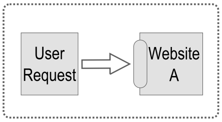

Figure 1: A common web request scenario

2 Foundations

2.1 Web Services

Today, Web Services [WCL+05] are the most prominent candidate to become the basic communication channel between two components of a distributed application. Founding on XML-based communication messages and formal service descriptions, the Web Service specifications are famous for their extensibility, ease of use, and last but not least their independence from programming languages and hardware platforms. Additionally, Web Services are considered as the most suitable realization of the Service-Oriented Architecture (SOA, [MLM+06]), and thus the number of Web Service technology adaptors and providers in industry, science and governments around the world increases continuously.

2.2 Service Composition and SOA

Typical web-based applications use an HTTP/HTML-based communication pattern, which allows human users to interact with a service in order to perform a specific task (e.g. buying a book in an online shop). In such scenarios, the server side implementation typically consists of a web server, an application logic component, and a database for storing persistent information. The client side usually makes a browser. This can be seen as some kind of one-on-one communication pattern: one human user interacts with one server system (see Fig. 1).

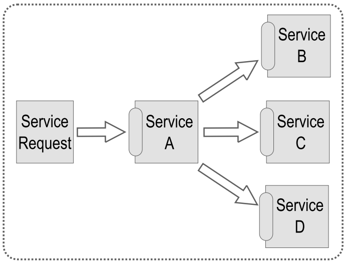

Figure 2: A service composition scenario

One of the most impressing advantages of the Web Services technology consists in the opportunity to completely automate the *client* side of such a one-on-one service invocation. This enables Web Service developers to embed calls to any external Web Services within their own applications. If those applications are provided as Web Services too, this is called a *service composition*. Fig. 2 shows a sample of such a service composition scenario, which can be classified as an one-to-many communication pattern: one user (not necessarily human) interacts with many Web Services, which may be spread over many servers or even companies.

Using the service composition approach, it becomes possible to easily *plug together* single Web Services into a service composition in order to perform some complex task, while the single Web Services still can be used by themselves. This architecture pattern is well-known as the *Service-Oriented Architecture*, and is an emerging and widespread paradigm for many computer system architectures around the world. Especially in business and science applications, the service-oriented approach is a premier choice for realizing any kind of workflows or even whole business processes.

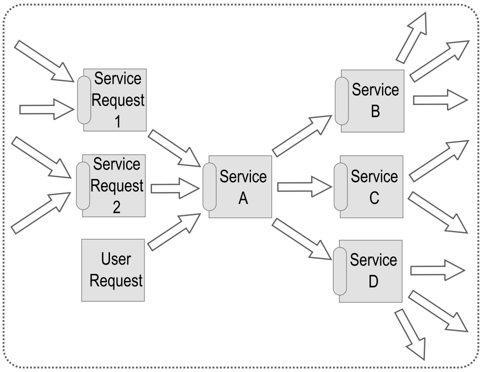

Figure 3: A full service-oriented architecture scenario

The communication pattern that results from this Service-Oriented Architecture approach is illustrated in Fig. 3. It may be classified as some kind of one-to-many-to-many-more communication, as a single Web Service request may trigger a huge set of service invocations, which again may cause further service invocations etc.

3 Attacks on Web Services

Due to their complex XML-based message structure, Web Service implementations need a lot more care—and thus computational resources—for parsing and processing incoming messages, compared to HTTP web servers. Further, due to their integration into service compositions, a fault within a single Web Service most likely causes failures within the service compositions that use it. Both will be discussed in the following sections.

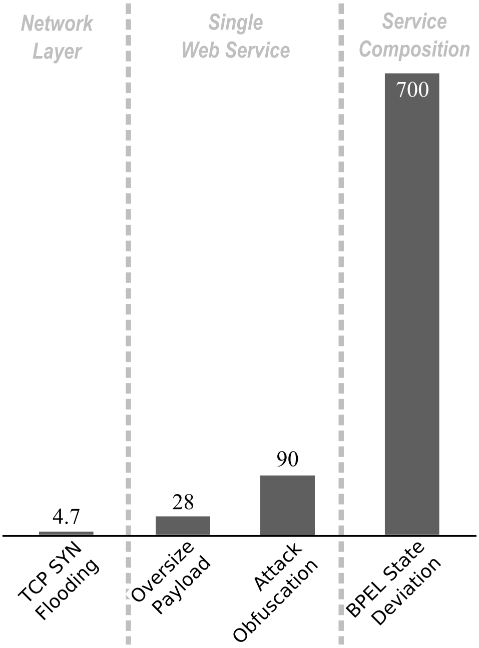

Figure 4: Caused memory consumption per message byte (cmp. [JGL08])

3.1 Single Web Services

Security researchers already found a bunch of attacks on single Web Services, reaching from integrity and confidentiality violations [MA05, GLS07] to highly severe Denial-of-Service attacks [JGHL07]. The actual and potential threat that was discovered within these analyses on single Web Services lead to an intense discussion on Web Service security, including countermeasure approaches [GL06] and Web Service security best practice instructions for adopters (e.g. [SWS07, MGMB07]).

3.2 Service Compositions

In service compositions, the number and impact severity of composition-specific attacks raises by far (see Fig. 4). Here, a typical Denial-of-Service attack even causes a full loss of availability for more than 2 hours, triggered by an attack with 500 KB of message traffic (the so-called *BPEL State Deviation* attack, [GJL07]).

Note that these evaluation values were measured for the pure Web Service frameworks only, and do not include the application-specific logic beneath the Web Service. Taking the application logic into account leads to a flood of new possibilities for attacks of any type, depending on the particular implementation (see [LHG08]).

3.3 Service-Oriented Architectures

Considering full Service-Oriented Architectures (like shown in Fig. 3), the potential threat that is caused by malicious or erroneous Web Service requests is tremendous. As an example, imagine an online store selling books. The online store provides an appropriate Web Service for placing book orders, payment is done via credit card. A valid request has to contain item informations on the ordered books, a shipping address and a credit card information. As the online store is not capable of validating credit card numbers itself, it delegates this task to appropriate Web Services provided by the particular credit card company—a typical Service-Oriented Architecture pattern.

In this scenario, a possible attack consists in flooding the online store's Web Service with nonsense messages that look like valid requests. Ordered books may be valid and in stock, but the credit card numbers used in the attack messages contain a random value. As a result, the online store will start to perform the full business process of book ordering for each single attack message. Obviously, the credit card validation Web Service will identify and reject the faulty credit card numbers, but up to that fault detection the attack message already caused a massive load for all other Web Services involved in the business process. Here, the input-output ratio of attack message size versus attack impact most likely will burst the scale of Fig. 4.

4 Future Work

As we have shown, attacks on single Web Services and basic service compositions have already become a serious threat to Web Services and SOA technology adopters. Evaluating the attack parameters of these known attacks, the potential threat for a full-featured service-oriented "Web Service Internet" is tremendous. As a consequence, it is necessary to intensify security research towards potential attack types, targets, countermeasures, interoperability and trust relationships in order to protect the business values involved in Web Service and SOA technology. We strongly recommend an intense examination of the potential security issues and countermeasure implications of these attacks, both in security research and SOA industry (cmp. [BGS$^+$07]).

To make a first step, we are going to concentrate on attack countermeasures, protection concepts and security enforcement for Web Services and basic service compositions. We intend to focus on attack detection and propagation facilities for service compositions, including the development of extensions to our Web Service security gateway [GL06] for the context of WS-BPEL-based service compositions.

Further, we work in enabling integrity and confidentiality for compositions of Web Services, along with practicability improvement of security enforcement within Web-Service-based service-oriented architecture

References

[BGS$^+$07] Johann Bizer, Rüdiger Grimm, Steffen Staab, Sebastian Meissner, Daniel Pähler, Christoph Ringelstein, Martin Rost, Jan Schallabök, and Felix Schwagereit. SOAinVO – Chancen und Risiken von Service-orientierten Architekturen in Virtuellen Organisationen. 2007.

[Bre07] Bill Brenner. Estonian attacks were a cyber riot, not warfare. *Black Hat*, 2007.

[GJL07] Nils Gruschka, Meiko Jensen, and Norbert Luttenberger. A Stateful Web Service Firewall for BPEL. *Proceedings of the IEEE International Conference on Web Services (ICWS 2007)*, 2007.

[GL06] Nils Gruschka and Norbert Luttenberger. Protecting Web Services from DoS Attacks by SOAP Message Validation. In *Proceedings of the IFIP TC-11 21. International Information Security Conference (SEC 2006)*, 2006.

[GLS07] Sebastian Gajek, Lijun Liao, and Jörg Schwenk. Breaking and fixing the inline approach. In *SWS '07: Proceedings of the 2007 ACM workshop on Secure web services*, pages 37–43, New York, NY, USA, 2007. ACM.

[JGHL07] Meiko Jensen, Nils Gruschka, Ralph Herkenhöner, and Norbert Luttenberger. SOA and Web Services: New Technologies, New Standards – New Attacks. In *Proceedings of the 5th IEEE European Conference on Web Services*, 2007.

[JGL08] Meiko Jensen, Nils Gruschka, and Norbert Luttenberger. The Impact of Flooding Attacks on Network-based Services. In *Proceedings of the IEEE International Conference on Availability, Reliability and Security*, 2008.

[LHG08] Lutz Lowis, Sebastian Höhn, and Maike Gilliot. Vulnerability Effect Propagation in Service-Oriented Architectures. *GI Sicherheit*, 2008.

[MA05] Michael McIntosh and Paula Austel. XML signature element wrapping attacks and countermeasures. In *SWS '05: Proceedings of the 2005 workshop on Secure web services*, pages 20–27, New York, NY, USA, 2005. ACM Press.

[MGMB07] Michael McIntosh, Martin Gudgin, K. Scott Morrison, and Abbie Barbir. Basic Security Profile Version 1.0. *Web Services Interoperability Organization (WS-I)*, 2007.

[MLM$^+$06] C. Matthew MacKenzie, Ken Laskey, Francis McCabe, Peter F Brown, and Rebekah Metz. Reference Model for Service Oriented Architecture 1.0. *OASIS Committee Specification*, 2006.

[Naz07] Jose Nazario. Estonian DDoS Attacks - A summary to date. *Arbor Networks Security Blog*, May 2007.

[Smi07] Adam Smith. Estonia: Under Siege on the Web. *Time Magazine*, 2007.

[SWS07] Anoop Singhal, Theodore Winograd, and Karen Scarfone. Guide to Secure Web Services. *Recommendations of the National Institute of Standards and Technology (NIST), Special Publication 800-95*, 2007.

[WCL$^+$05] Sanjiva Weerawarana, Francisco Curbera, Frank Leymann, Tony Storey, and Donald F. Ferguson. *Web Services Platform Architecture: SOAP, WSDL, WS-Policy, WS-Addressing, WS-BPEL, WS-Reliable Messaging, and More*. Prentice Hall PTR, 2005.