Blame Wars

How To Attribute Responsibility

SANS Pen Test Hackfest Europe Summit, Berlin, 2019/7/22

David Fuhr
Blame Wars
How To Attribute Responsibility
SANS Pen Test Hackfest Europe Summit, Berlin, 2019/7/22
David Fuhr
David Fuhr
Head of Research, HiSolutions AG

• Maths
• InfoSec
• Gestalt/Coaching

• Twitter: @0xdhf
• research.hisolutions.com
Attribution Is Hard
Blaming Is Easy

NAME

git-blame - Show what revision and author last modified each line of a file
Responsibility Is Harder
November, 2018
“Responsible” Coordinated Disclosure

“The German government-issued identity card (nPA) allows German citizens to not only prove their identity in person, but also against online services (by using the embedded RFID chip). A critical security vulnerability in the Governikus Autent SDK enables an attacker to impersonate arbitrary users against affected web applications.”

German nPA – „New“ (e)*ID Card
Come Get SAML! Governikus Autent S-D-K

SAML Web Browser SSO

Tom Scavo, InCommon
eID Auth

1. web app redirects to https://127.0.0.1:24727/eID-Client? tcTokenURL=https://localhost/AutentSAMLDemo/NewRequesterServlet

2. eID client initiates authentication communicating with web app, auth server, and RFID chip / reader

3. eID client sends SAML response containing the data read from the id card to web app via HTTP GET: https://<host>/<path>?SAMLResponse=...&Signature=...

4. eID client redirects (HTTP 302) back to web app. The user is now authenticated.
Attack

- Governikus Autent SDK offers method `HttpRedirectUtils.checkQueryString` to parse query and verify signature. Afterwards, SAML response is parsed.

- `HttpRedirectUtils.checkQueryString` creates a canonical version of the query string using the last occurrence of a parameter.

- `HttpServletRequest.getParameter` according to Java Servlet specification always yields the first parameter of a given name.

- If an attacker supplies multiple parameters named `SAMLResponse` the signature is verified against last occurrence, while the SAML response that is processed further is taken from the first occurrence.
Exploit Scenarios

- Attack **requires one valid query string** signed by the auth server, **no matter for whom or when the signature was issued.**
- In practice, shared auth servers are often used (e.g. eid1.eid-service.de at least by BStU, German Justice Department and the German Pension Fund).
- Although a query string is only valid for a short time, **expiration is checked on the manipulated data.**

Where to get a signed query?

- Auth with your own id card. Beware: Your real ID data appears in access log
- Logs from servers or from the eID client
- Google dorks: “richclient_info.log SAMLResponse”, “AusweisApp2 filetype:log”, ...
Attack Restrictions

“[…]

Moreover, this attack is only applicable to web applications that use the method **HttpServletRequest.getParameter**. If an application rejects multiple parameters with the same name, the vulnerability cannot be exploited. A web application firewall (WAF) configured to reject multiple parameters of the same name would also prevent this attack.”

This information is intended to clarify the current coverage (November 2018) concerning a vulnerability in the Governikus Autent Software Development Kit (SDK) version 3.8.1 and the resulting conclusions drawn from the press about insecure implementations of the eID function of the German ID card:

Last year, Governikus KG presented a demo sample based on the Governikus Autent SDK, in the context of a Governikus AusweisApp2 user forum, to illustrate the straightforward and rapid implementation of the eID function of the German ID card for service providers. This demo example was also publicly available for download without including the full Governikus Autent SDK.

SEC Consult has used this demo scenario, which at no time was expressly intended to carry out a full implementation in real operation, to check for vulnerabilities and in July of the previous year reported a vulnerability to the CERT-Bund.

Governikus Statement 2/4

The attack scenario, described by SEC Consult, makes use of the fact that in the demo scenario of the Autent SDK, the inspection of a “SAMLResponse” performs correctly, but then a further attached parameter by the name “SAMLResponse” is processed.

While the check routine origins from the Autent SDK, processing the SAML response is a standard call to the servlet API (getParameter) and is thus part of the example implementation and has nothing to do with the Autent SDK.

The demo scenarios included in the Autent SDK are intended to demonstrate the libraries, as the name implies. However, they did not and do not claim to mention or even implement further security measures that are required in real operation. These must be taken by the respective service providers or operators of such services, for example to protect against XSS attacks, SQL injection, replay attacks, etc. These URL inspections are the measure that prevents the attack.
Here, the open “SAML Web Browser SSO Profile” protocol (Redirect Binding) specified by OASIS is used. The SAML protocol is deliberately open so that further parameters can be transmitted, if needed by the respective application. The inspection rules for the service provider arise among others from the standard itself (https://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf, section 3.4.4.1 “(...) the relying party MUST ensure that the parameter values to be verified are [ordered] as required by the signing rules above“). The measures to be taken result from OASIS and OWASP recommendations, which are taken into account by our customers as part of the integration and are cited as a prerequisite for their implementation. In this case, especially the recommendations of OWASP are to be considered (https://www.owasp.org/index.php/Input_Validation_Cheat_Sheet).

Nevertheless, additional measures have to be implemented on an application-related basis.
We agree with the assessment that an example is often taken over unintentionally, even if the included readme file indicates that it is a demo, and besides that only runs on localhost. That is why we included a quick-fix in the corresponding routine in August 2018 that returns an error if relevant URL parameters (SigAlg, SAMLResponse, RelayState) actually occur more than once. **We have made this update available to our customers immediately.** This content inspection is indeed carried out in real scenarios prior to processing within the SDK, by implementing the OASIS and OWASP recommendations. Of course, the SDK only runs in Java environments and the online identification function is just one possible application.

Furthermore, the SDK is a product of the Governikus KG. Therefore, it cannot act on behalf of the online identification function.

The conclusion drawn up by both, SEC Consult and some of the journalists that all integrations of the eID function implemented by Autent would be insecure, is quite honestly wrong.
SEC Consult Update 2018-11-26

In an official statement and on Twitter, Governikus KG recently suggested that SEC Consult and media outlets misrepresented a vulnerability that has been published in our blog. As the SEC Consult Vulnerability Lab always strives to be as precise and accurate as possible, including all public communication, Governikus received both the technical advisory and the blog post beforehand (see “Vendor Contact Timeline” in our advisory).

Our security consultants have been in contact with Governikus via CERT-Bund since July 2018 and incorporated Governikus’ feedback into the final version of our blog post before it was released. Up to this point we experienced the communication with Governikus to be very professional and solution oriented.

As Governikus states that the attack is not possible with real applications, we want to disclose the circumstances of the vulnerability identification: The mentioned authentication bypass vulnerability was identified during a 3rd party project of an internet-facing web application. In order to simplify communication with both Governikus and the public, we opted to replicate the vulnerability with a publicly available software component (i.e. the Governikus Demo application which was publicly available then) instead of the actual implementation of the 3rd party.

Through that, we are aware that at least this application and the demo application were indeed vulnerable to the attack. We did not (and would never) conclude in our publication that all applications that facilitate the Governikus Autent SDK are vulnerable. [...]

21
Christian Kahlo @ckahlo

Ich bin seit 2009 in die Personalausweis Entwicklung involviert, bin in den BSI-Arbeitsgruppen, habe eID-Server, eID-Clients, Ausweise (Chip) implementiert. Die Meldung zum "Ausweishack" ist ein gnadenlos aufgeblasener Fake. @thealuc @hanno @sec_consult @heisec @ettisan

Translated from German by Microsoft
I have been involved in the development of the IDENTITY card since 2009, am in the BSI working groups, have implemented eID servers, eID clients, ID cards (chip). The message about the "card hack" is a mercilessly inflated fake. @thealuc @hanno @sec_consult @heisec @ettisan

1:27 PM - 26 Nov 2018
1 Like
Tweet your reply

Christian Kahlo @ckahlo · 26 Nov 2018

Nichts im Chip, im eID-Server, der Infrastruktur, der App ist kaputt. Es wurde lokal eine Demo-App für das SAML-Protokoll in der Test-PKI verborgen. Die meisten hingegen verwenden gar kein SAML wegen seiner Schwächen. Anfällig war diese Eigenbau SAML Anbindung, nicht der nPA.

Translated from German by Microsoft
Nothing in the chip, in the eID server, in the infrastructure, in the app is broken. A demo app for the SAML protocol in the test PKI was bent locally. Most, on the other hand, do not use SAML at all because of its weaknesses. Vulnerable was this self-construction SAML connection, not the nPA.

12:39 PM - 26 Nov 2018
3 Likes
Tweet your reply

Aluc @thealuc · 26 Nov 2018

Replying to @ckahlo
mehr hat der ja auch nicht behauptet, so wie ich das verstanden habe

Translated from German by Microsoft
he didn’t say more, as I understood it

3 Likes
As a company, we are **always grateful for checking our applications**, especially with regard to possible security vulnerabilities. Since this gap became known, we have been in close contact with SEC Consult. Accordingly we **would like to thank SEC Consult for the good and professional cooperation!** This text, in turn, lacks the usual quality of Golem texts. From our point of view, the productive exchange by e-mail from which you quote is not reflected in this article. **Your questions were answered in detail several times.** Basically, we would like to emphasize that critical questions from a journalistic point of view and exaggerations as stylistic devices are expressly welcome. However, if they are used to avoid having to familiarise oneself with a topic that requires some explanation, then this is pure **laziness.** It is particularly **annoying** when it comes to well-known standards that are presented in "two longer documents" on around 35 pages. A Google search could have filled this knowledge gap problem-free and in seconds - perhaps you would have stumbled thereby even on articles of Golem.de. However, to assume a refusal attitude already borders on **Chuzpe.**

Best regards from Bremen

Petra Waldmüller-Schantz Communications Manager Governikus KG

Author: jg (Golem.de) 22.11.18 - 10:35

**Dear Mrs. Waldmüller-Schantz, the author of this article will contact you directly for clarification. MfG Juliane Gunardono**
Your Host JON PERTWEE

WHODUNNIT?

THE MURDER MYSTERY GAMESHOW

2 Disc Set

THE COMPLETE SECOND SERIES
Supply Chain
„Supply“ „Chain“
„Supply“

A TATTOO PEN
FOR EVERY PENTESTER
Alice (and Bob) in Supply Chains

Security Documentation
Security Guidelines
Security Support

Product Development
Product Supplier

Integration / Commissioning
System integrator

Operation & Maintenance
Asset Owner

Requirements

IEC 62443-1-1
Machinery Directive 1/3

- (14) The **essential health and safety requirements should be satisfied** in order to ensure that machinery is safe; these requirements should be applied with discernment to take account of the state of the art at the time of construction and of technical and economic requirements.
- (15) **Where the machinery may be used by a consumer, that is to say, a non-professional operator, the manufacturer should take account of this in the design and construction. The same applies where a machine is normally used to provide a service to a consumer.**
Machinery Directive 2/3

1.1.2. Principles of safety integration

(a) Machinery must be designed and constructed so that it is fitted for its function, and can be operated, adjusted and maintained without putting persons at risk when these operations are carried out under the conditions foreseen but also taking into account any reasonably foreseeable misuse thereof.

The aim of measures taken must be to eliminate any risk throughout the foreseeable lifetime of the machinery including the phases of transport, assembly, dismantling, disabling and scrapping.

(b) In selecting the most appropriate methods, the manufacturer or his authorised representative must apply the following principles, in the order given:

— eliminate or reduce risks as far as possible (inherently safe machinery design and construction),
— take the necessary protective measures in relation to risks that cannot be eliminated,
— inform users of the residual risks due to any shortcomings of the protective measures adopted, indicate whether any particular training is required and specify any need to provide personal protective equipment.
(c) When designing and constructing machinery and when drafting the instructions, the manufacturer or his authorised representative must envisage not only the intended use of the machinery but also any reasonably foreseeable misuse thereof.

The machinery must be designed and constructed in such a way as to prevent abnormal use if such use would engender a risk. Where appropriate, the instructions must draw the user's attention to ways — which experience has shown might occur — in which the machinery should not be used.

(e) Machinery must be supplied with all the special equipment and accessories essential to enable it to be adjusted, maintained and used safely.
Machinery Directive
Product Liability

- regardless of guilt / negligence
- for 10 years
- anyone in the supply chain can be sued!
- 85 M€ limit
- Death or injury & damage to items for personal use/consumption → no business damage
- Except developing risks (“state of science and technology”) – also for detecting defects
- Software *is* covered, but
  - services (*aaS) out of scope
  - hard to prove: “The injured person shall be required to prove the damage, the defect and the causal relationship between them.”
- hard to enforce: complex supply chains, open source etc.
Product Liability
Doing things the quick and dirty way sets us up with a technical debt, which is similar to a financial debt. Like a financial debt, the technical debt incurs interest payments, which come in the form of the extra effort that we have to do in future development because of the quick and dirty design choice. We can choose to continue paying the interest, or we can pay down the principal by refactoring the quick and dirty design into the better design. – Martin Fowler, 2009 [https://martinfowler.com/bliki/TechnicalDebtQuadrant.html](https://martinfowler.com/bliki/TechnicalDebtQuadrant.html)
Security Debt Is (the New) Technical Debt

Security debt, the lack of proper testing techniques and security preparedness, creates long-term costs companies are merely putting off to the future. Security fails in the short term and the costs snowball in the long term.

https://www.nowsecure.com/blog/2015/10/08/security-debt-is-the-new-technical-debt/
Technical Debt

```
Reckless

“We don’t have time for design”

Prudent

“We must ship now and deal with consequences”

Deliberate

Inadvertent

“What’s Layering?”

“Now we know how we should have done it”
```

Martin Fowler, 2009

https://martinfowler.com/bliki/TechnicalDebtQuadrant.html
Collateralized Technical Debt

- Collateralized Debt Obligation (CDO) -> CTO
- CDO/CTO Squared
- Weapons of Technical Mass Destruction -- Warren Buffet Overflow
Sheep Happens – Shame

2 Types of Shame

Neurotic    Existential
The most paradoxical aspect of neurotic shame is that it is the core motivator of the superachieved and the underachieved, the star and the scapegoat, the righteous and the wretched, the powerful and the pathetic. – John Bradshaw
Existential Shame
Existential Skills

- Self-awareness, acceptance, and acknowledgement of one’s failures and limitations
- Self-forgiveness and self-compassion
- Self-affirmation through re-authoring and reframing
- Self-affirming the intrinsic value and sacredness of life
- Taking personal responsibility for positive change
- Seeking a support group
- Focusing on the meaning people attach to events
- Seeking deeper meaning and self-transcendence
- Seeking deeper understanding through verbalizing one’s emotions and dreams

Paul Wong: From Shame to Wholeness: *An Existential Positive Psychology Perspective*
git-blame-someone-else

"I love git-blame-someone-else!!" -Linus Torvalds says

Install

$ git clone https://github.com/jayphelps/git-blame-someone-else.git
$ cd git-blame-someone-else
$ sudo make install

Usage

$ git blame-someone-else <author> <commit>
git-praise

Sometimes you find the most amazing piece of code and want to know who to thank for this wonderful contribution and then it feels wrong to use `git blame`.

This is why `git praise` exists, just praise that file and you’ll find out who to thank!

Usage

The syntax is exactly the same as for `git blame`.

Try doing `git praise path/to/some/file` or `git help blame` for all options.