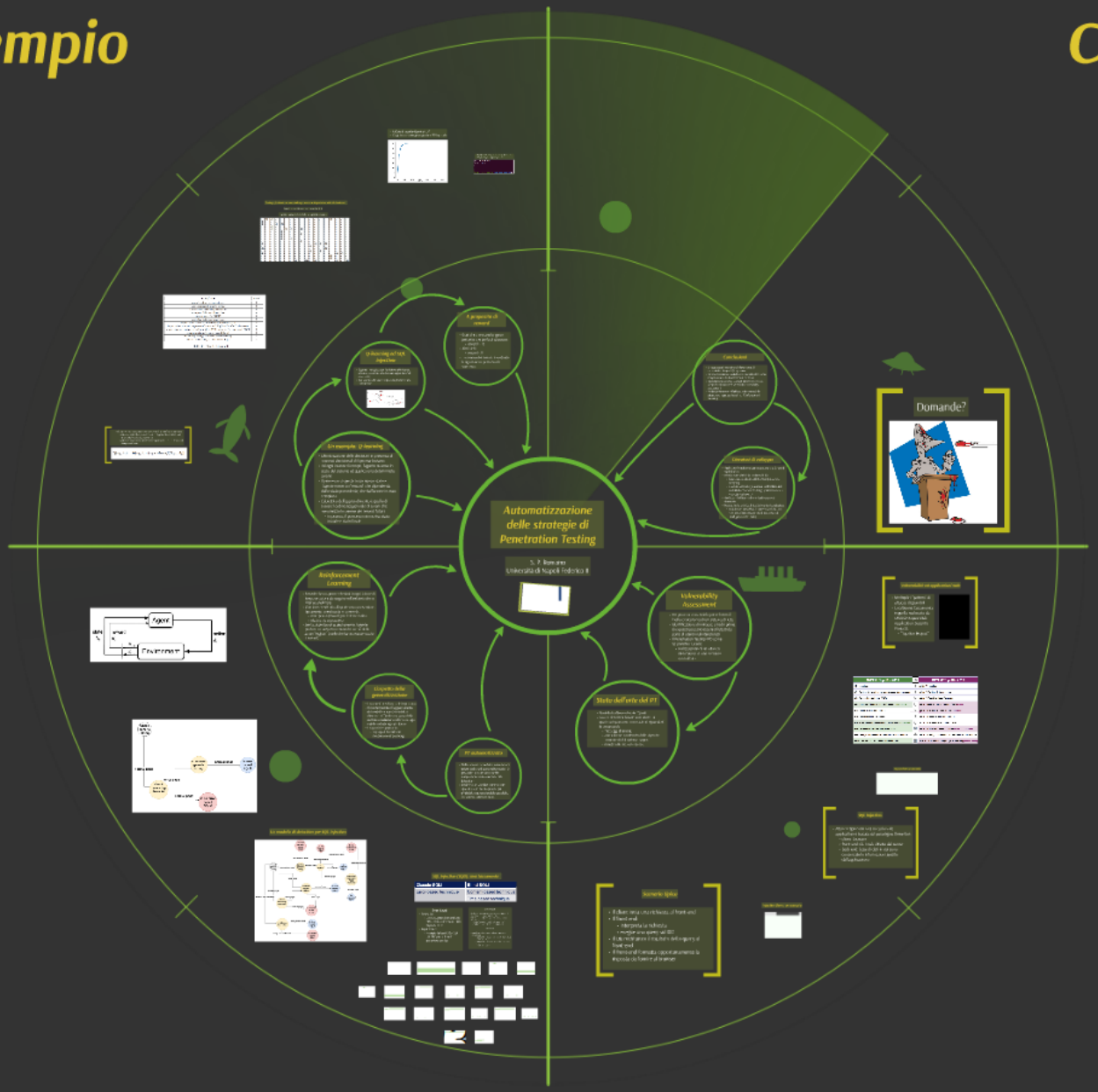
Caso di esempio

Conclusioni



Contributo

Contesto

WORKSHOP GARR 2019
NET MAKERS
LA COMUNITÀ CHE INNOVA LA RETE

Prezi

Automatizzazione delle strategie di Penetration Testing

S. P. Romano
Università di Napoli Federico II

About me

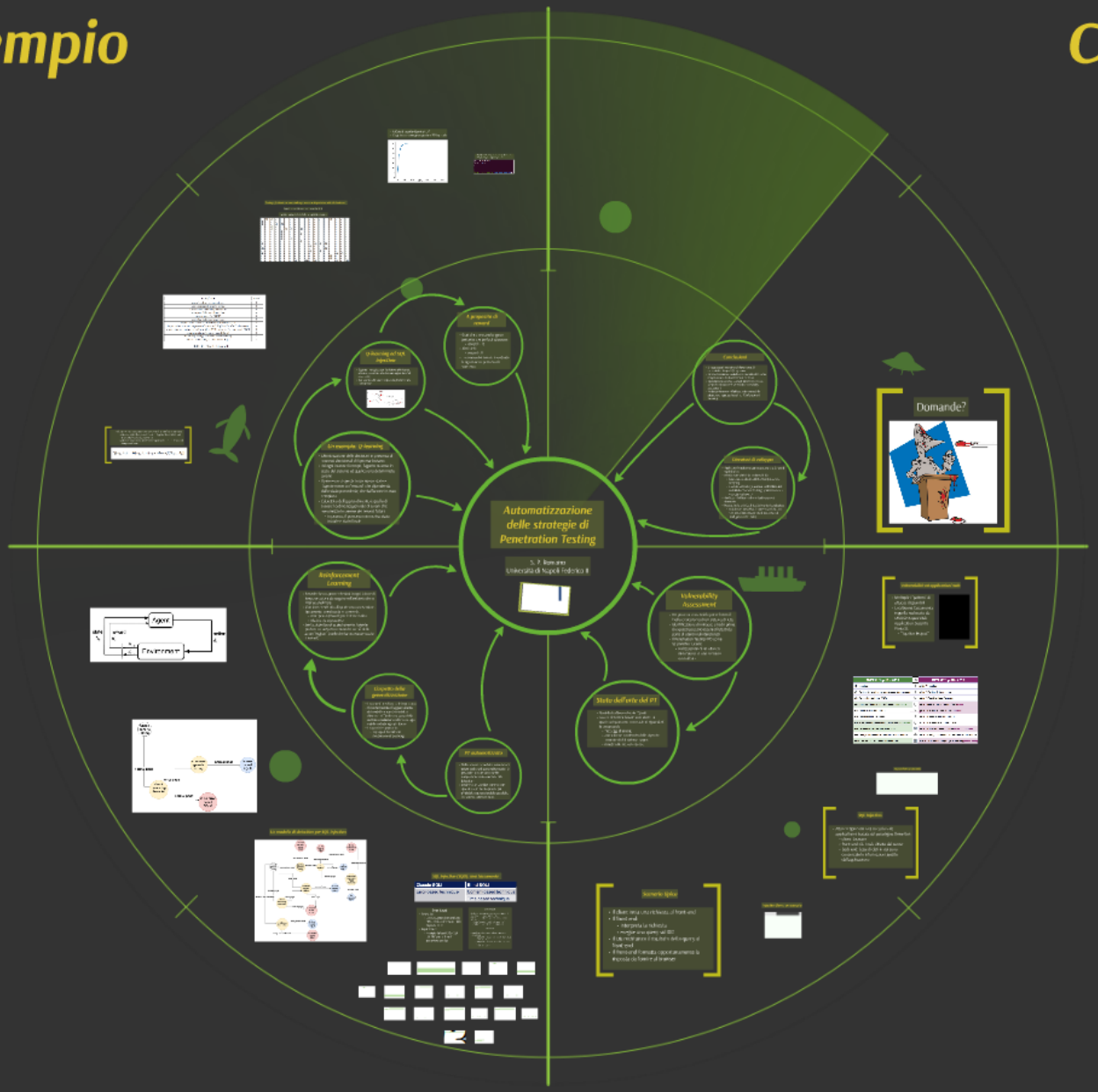
- Professor at Federico II University in Napoli:
  Teaching Computer Networks, Telematic Applications and Network Security
- Managing Director at the Apple Developer Academy in Napoli
- Tech transfer addicted:
  www.epsilonline.com —> cloud, web apps, microservices, etc.
  www.meetecho.com —> real-time multimedia applications
  www.cybersecsi.com (coming soon!) —> network security

spromano@unina.it

@spromano

Caso di esempio

Conclusioni



Contributo

Contesto

WORKSHOP GARR 2019
NET MAKERS
LA COMUNITÀ CHE INNOVA LA RETE

Prezi

Vulnerability Assessment

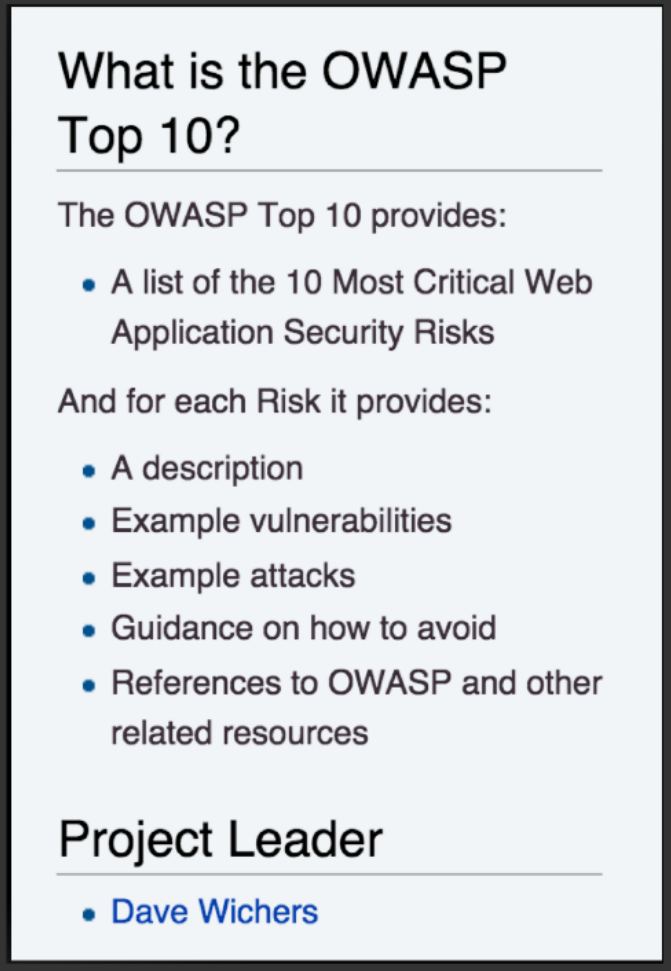
- Un processo essenziale per valutare il livello di sicurezza di un sistema di rete
- Identificazione di minacce e rischi prima che questi possano essere sfruttati da parte di utenti malintenzionati
- Il Penetration Testing (PT) come strumento cruciale:
  - realizzazione di un attacco informatico in uno scenario controllato

Assessment

- Un processo essenziale per valutare il livello di sicurezza di un sistema di rete
- Identificazione di minacce e rischi prima che questi possano essere sfruttati da parte di utenti malintenzionati
- Il Penetration Testing (PT) come strumento cruciale:
  - realizzazione di un attacco informatico in uno scenario controllato

Vulnerabilità ed applicazioni web

- Molteplici "pattern" di attacco disponibili
- Una buona tassonomia è quella realizzata da OWASP (Open Web Application Security Project):
  - "Top Ten Project"



What is the OWASP Top 10?

The OWASP Top 10 provides:

- A list of the 10 Most Critical Web Application Security Risks

And for each Risk it provides:

- A description
- Example vulnerabilities
- Example attacks
- Guidance on how to avoid
- References to OWASP and other related resources

Project Leader

- Dave Wichers

n" di

omia
da
b
ty

ct"

Prezi

What is the OWASP Top 10?

The OWASP Top 10 provides:

- A list of the 10 Most Critical Web Application Security Risks

And for each Risk it provides:

- A description
- Example vulnerabilities
- Example attacks
- Guidance on how to avoid
- References to OWASP and other related resources

Project Leader

- Dave Wichers

OWASP Top 10 - 2013		OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

Injection flaws: caratteristiche

Threat Agents	Attack Vectors	Security Weakness		Technical Impacts	Business Impacts
Application Specific	**Exploitability EASY**	**Prevalence COMMON**	**Detectability AVERAGE**	**Impact SEVERE**	**Application / Business Specific**
Consider anyone who can send untrusted data to the system, including external users, internal users, and administrators.	Attacker sends simple text-based attacks that exploit the syntax of the targeted interpreter. Almost any source of data can be an injection vector, including internal sources.	Injection flaws occur when an application sends untrusted data to an interpreter. Injection flaws are very prevalent, particularly in legacy code. They are often found in SQL, LDAP, Xpath, or NoSQL queries; OS commands; XML parsers, SMTP Headers, program arguments, etc. Injection flaws are easy to discover when examining code, but frequently hard to discover via testing. Scanners and fuzzers can help attackers find injection flaws.		Injection can result in data loss or corruption, lack of accountability, or denial of access. Injection can sometimes lead to complete host takeover.	Consider the business value of the affected data and the platform running the interpreter. All data could be stolen, modified, or deleted. Could your reputation be harmed?

SQL Injection

- Attacco tipico nei casi in cui la web application è basata sul paradigma three-tier:
  - client: browser
  - front-end: GUI web offerta dal server
  - back-end: base di dati in cui sono conservate le informazioni gestite dall'applicazione

Scenario tipico

- Il client invia una richiesta al front-end
- Il front-end:
  - interpreta la richiesta
  - esegue una query sul DB
- Il DB restituisce il risultato della query al front-end
- Il front-end formatta opportunamente la risposta da fornire al browser

Injection flaws: un esempio

Scenario #1: The application uses untrusted data in the construction of the following **vulnerable** SQL call:

```
String query = "SELECT * FROM accounts WHERE custID='" +
request.getParameter("id") + "'";
```

Scenario #2: Similarly, an application's blind trust in frameworks may result in queries that are still vulnerable, (e.g., Hibernate Query Language (HQL)):

```
Query HQLQuery = session.createQuery("FROM accounts WHERE
custID='" + request.getParameter("id") + "'");
```

In both cases, the attacker modifies the 'id' parameter value in her browser to send: ' or '1'='1. For example:

```
http://example.com/app/accountView?id=' or '1'='1
```

This changes the meaning of both queries to return all the records from the accounts table. More dangerous attacks could modify data or even invoke stored procedures.

Scenario #1: The application uses untrusted data in the construction of the following **vulnerable** SQL call:

```
String query = "SELECT * FROM accounts WHERE custID='" +
request.getParameter("id") + "'";
```

Scenario #2: Similarly, an application's blind trust in frameworks may result in queries that are still vulnerable, (e.g., Hibernate Query Language (HQL)):

```
Query HQLQuery = session.createQuery("FROM accounts WHERE
custID='" + request.getParameter("id") + "'");
```

In both cases, the attacker modifies the 'id' parameter value in her browser to send: ' or '1'='1. For example:

```
http://example.com/app/accountView?id=' or '1'='1
```

This changes the meaning of both queries to return all the records from the accounts table. More dangerous attacks could modify data or even invoke stored procedures.

Stato dell'arte del PT

- Modelli di attacco ("exploit") noti
- Fase di detection basata su tentativi ai quali corrispondono osservazioni riguardanti la presenza di:
  - messaggi di errore;
  - anomalie nel contenuto delle risposte provenienti dal sistema target;
  - ritardi nelle risposte stesse.

Stato dell'arte del PT

- Modelli di attacco ("exploit") noti
- Fase di detection basata su tentativi ai quali corrispondono osservazioni riguardanti la presenza di:
  - messaggi di errore;
  - anomalie nel contenuto delle risposte provenienti dal sistema target;
  - ritardi nelle risposte stesse.

PT automatizzato

- Dalla osservazione della sequenza di azioni svolte dal penetration tester è possibile ricavare un modello comportamentale orientato alla detection;
- modello che sarebbe interessante riprodurre, in modo quanto più affidabile e generalizzabile possibile, in maniera automatizzata.

- Dalla osservazione della sequenza di azioni svolte dal penetration tester è possibile ricavare un modello comportamentale orientato alla detection;
- modello che sarebbe interessante riprodurre, in modo quanto più affidabile e generalizzabile possibile, in maniera automatizzata.

SQL injection (SQLi): una tassonomia

Classic SQLi	Blind SQLi
Error-based technique	Content-based technique
	Time-based technique

Error based

- Detection:
  - Un messaggio di errore lato DB è cablato all'interno della risposta HTTP
- Exploitation:
  - Impiego del costrutto "SQL UNION" per la fase di estrazione dei dati

Content based

- Disallineamento rispetto ad una normale risposta HTTP (es., rimozione di uno o più tag HTML)
- Il risultato della query viene stimato in base al tipo di risposta:
  - Pagina "normale": risultato della query "vero"
  - Pagina "anomala": risultato della query "falso"

Time based

- Tempo di risposta in corrispondenza di query SQL "temporizzate"
  - tempo di risposta compatibile con il valore atteso:
    - risultato "vero"
  - tempo di risposta non compatibile con il valore atteso:
    - risultato "falso"

Error based

- Detection:
  - Un messaggio di errore lato DB è cablato all'interno della risposta HTTP
- Exploitation:
  - Impiego del costrutto "SQL UNION" per la fase di estrazione dei dati

- Disa
(es.
- Il ris
risp
  -
  -
- Tem
"ter

Content based

- Disallineamento rispetto ad una normale risposta HTTP (es., rimozione di uno o più tag HTML)
- Il risultato della query viene stimato in base al tipo di risposta:
  - Pagina "normale": risultato della query "vero"
  - Pagina "anomala": risultato della query "falso"

Time based

- Tempo di risposta in corrispondenza di query SQL "temporizzate"
  - tempo di risposta compatibile con il valore atteso:
    - risultato "vero"
  - tempo di risposta non compatibile con il valore atteso:
    - risultato "falso"

Figure 4.6: Registration form

```
GET http://s101627-101522-3ay.sipontum.hack.me/index.php?act=sukses HTTP/1.1
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:67.0) Gecko/20100101 Firefox/67.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: it-IT,it;q=0.8,en-US;q=0.5,en;q=0.3
Referer: http://s101627-101522-3ay.sipontum.hack.me/index.php
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: s101627-101522-3ay.sipontum.hack.me
Cookie: __utmz=233483271.1559550762.28.6.utmcsr; c_login=
YToyOntzOjQ6InVzZXIiO3M6NDoiZG1vcyI7czo5OiJ1c2VyX3R5cGUiO3M6NDoidXNlciI7fQ%3D%3D; __unam=657356c-
16a12180ba2-4e47c27-264; __utmt=1; __utmb=233483271.7.10.1561715551; __utmc=233483271; __utma=233483271
.1398831352.1555081643.1561386818.1561715551.33
```

Figure 4.7: HTTP cookie

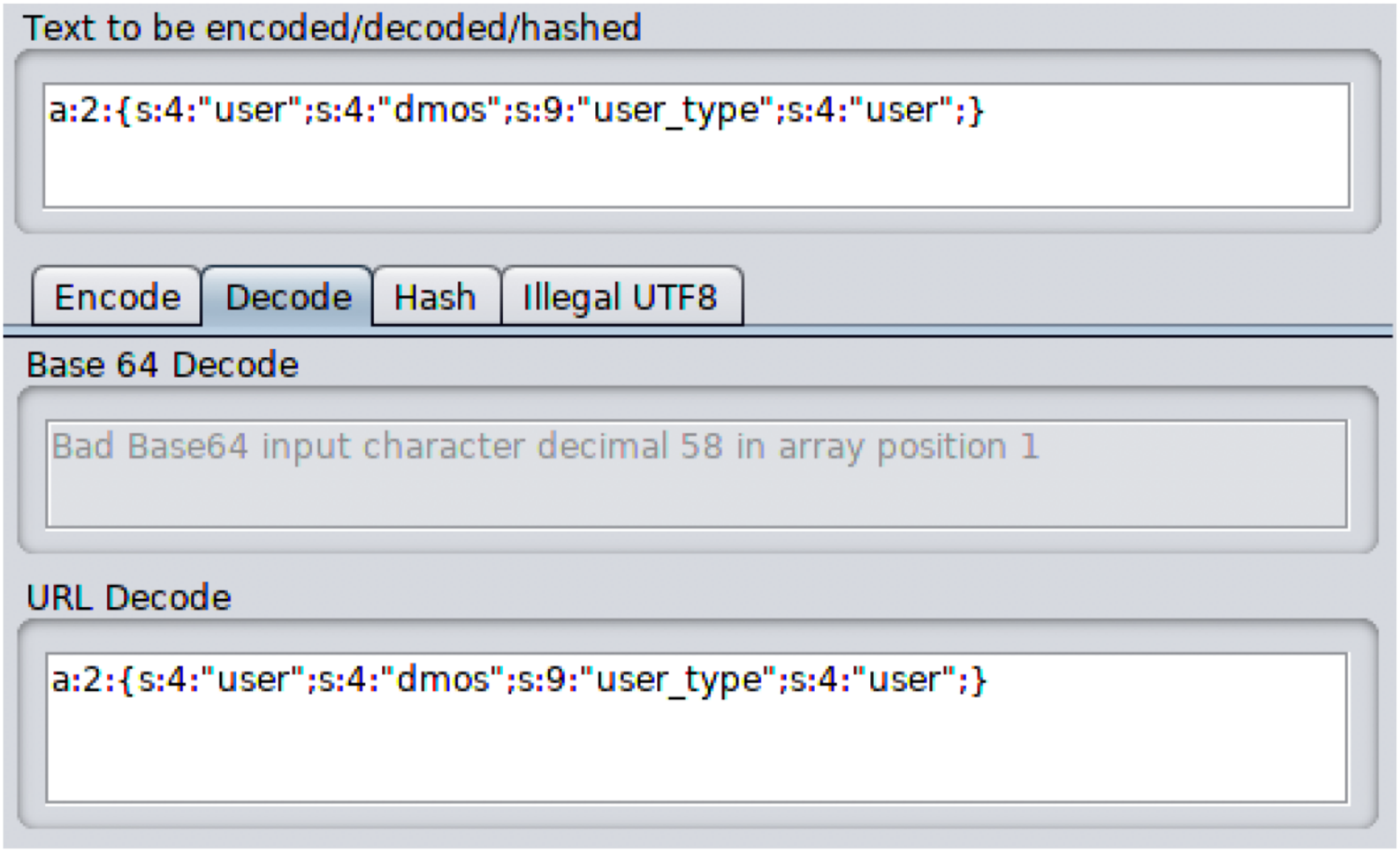


Figure 4.8: Cookie content

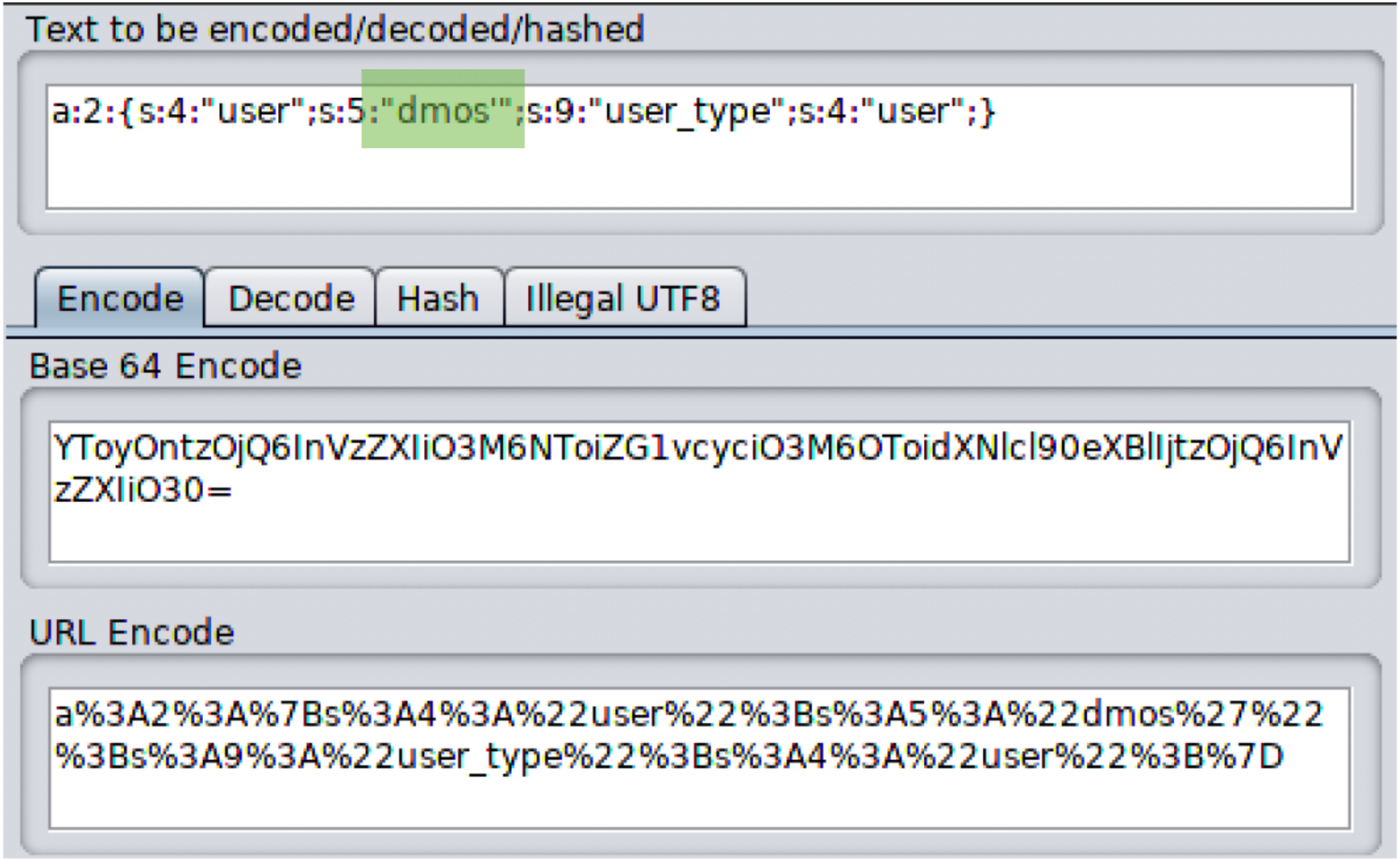


Figure 4.9: Most efficient path:apex

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Microsoft-IIS/7.5
X-Allocated-On-The-Fly-By: Coliseum Web Application Security Framework
X-Powered-By: Caendra Team
Date: Fri, 28 Jun 2019 10:25:13 GMT
Content-Length: 263
```

```
login successfuly<br><br><br />
<b>Warning</b>:  mysql_numrows() expects parameter 1 to be resource, boolean
given in <b>C:\inetpub\wwwroot\coliseum\sandboxes\101627-101522\BODY\inner\
index.php</b> on line <b>89</b><br />
<a href="index.php?act=logout">Logout</a>
```


Figure 4.10: Response to apex injection

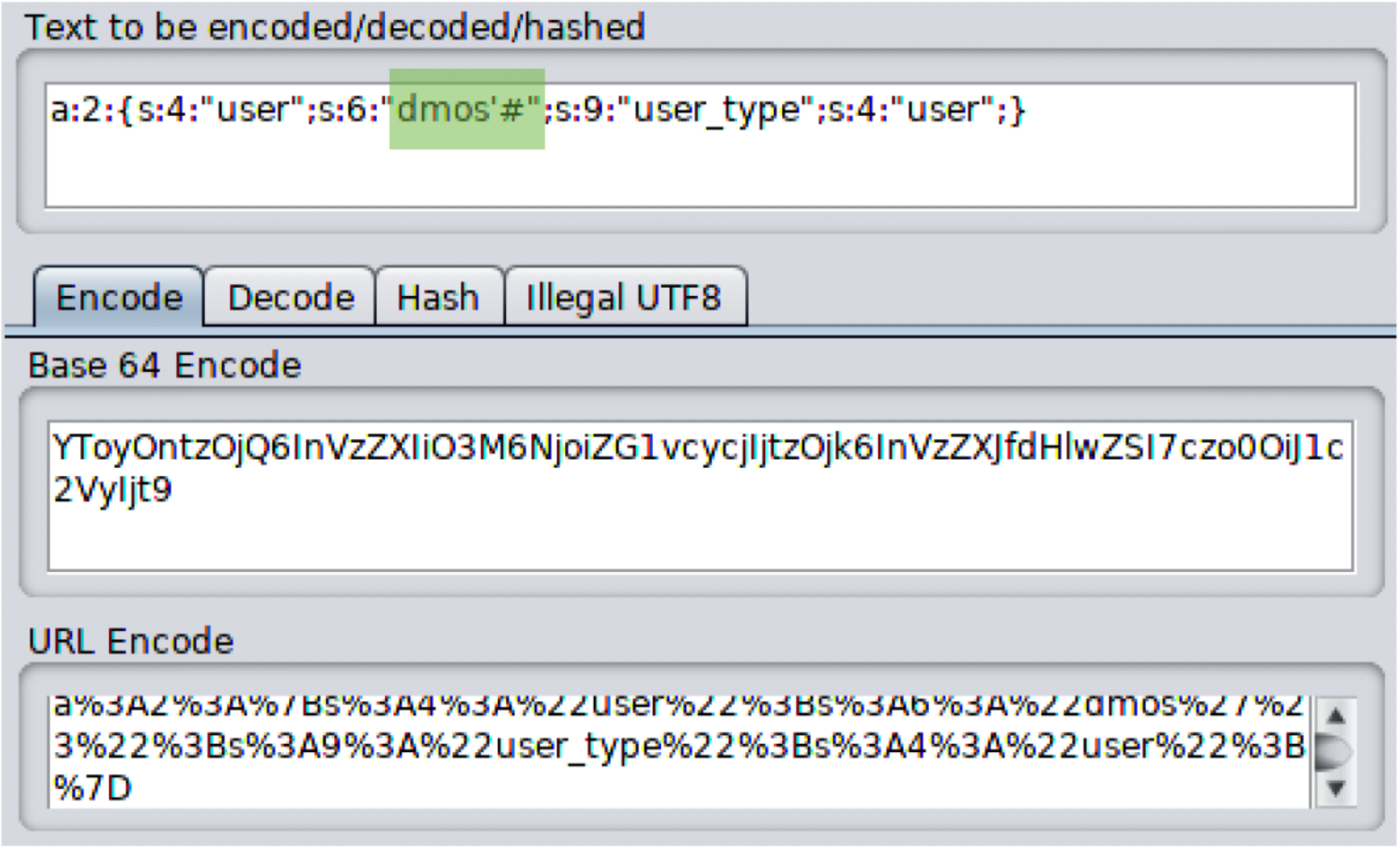


Figure 4.11: Most efficient path:hash comment

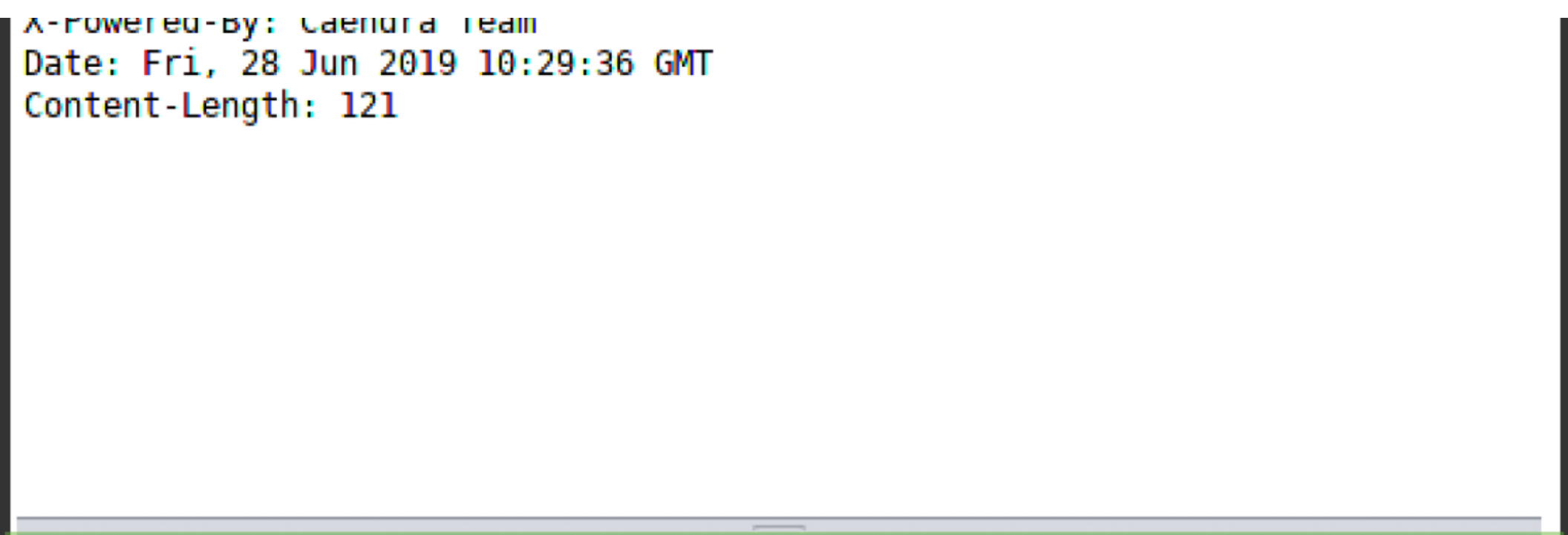


Figure 4.12: Response to hash comment

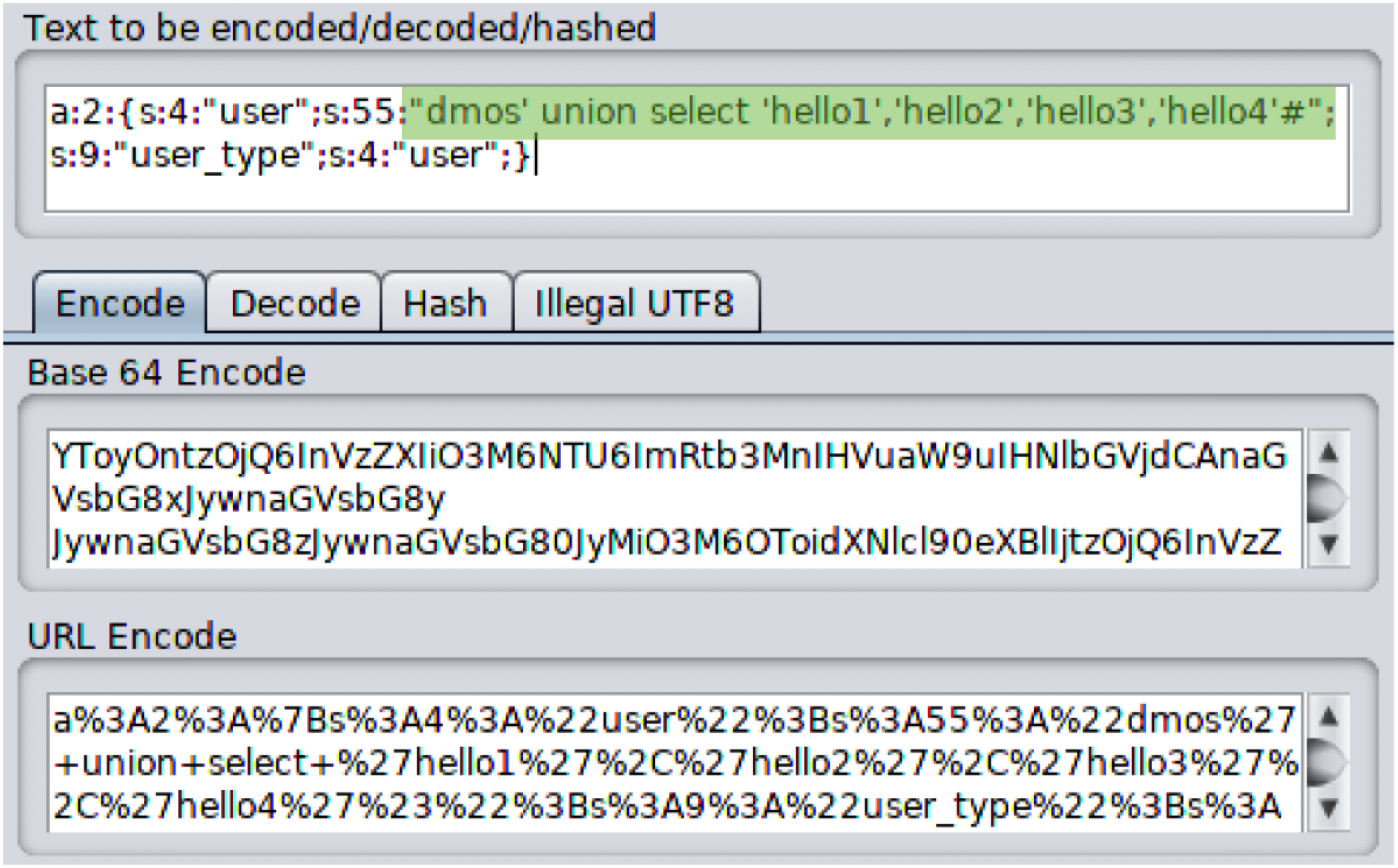


Figure 4.13: Column number discovery

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Microsoft-IIS/7.5
X-Allocated-On-The-Fly-By: Coliseum Web Application Security Framework
X-Powered-By: Caendra Team
Date: Fri, 28 Jun 2019 10:47:36 GMT
Content-Length: 180
```

```
login successfuly<br><br>Hi <b>dmos</b><h1>Sorry, You are just an user :( !<
/h1>Hi <b>hello2</b><h1>Sorry, You are just an hello4 :( !</h1><a href="
index.php?act=logout">Logout</a>
```

Figure 4.14: Column in the response

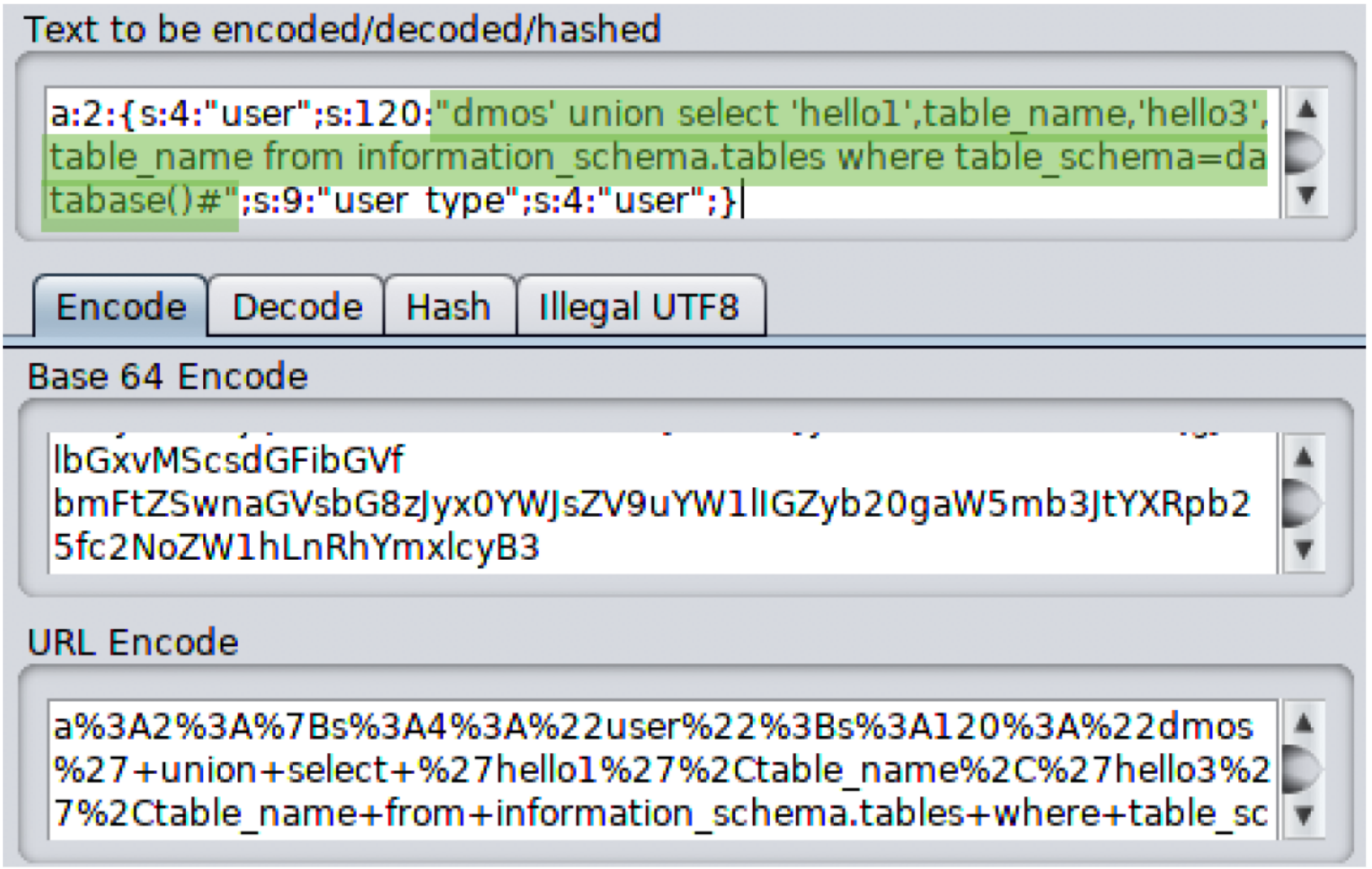


Figure 4.15: Discovery table name

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Microsoft-IIS/7.5
X-Allocated-On-The-Fly-By: Coliseum Web Application Security Framework
X-Powered-By: Caendra Team
Date: Fri, 28 Jun 2019 10:51:31 GMT
Content-Length: 182
```

```
login successfuly<br><br>Hi <b>dmos</b><h1>Sorry, You are just an user :( !<
/h1>Hi <b>anggota</b><h1>Sorry, You are just an anggota :( !</h1><a href="
index.php?act=logout">Logout</a>
```

Figure 4.16: Table name in the response

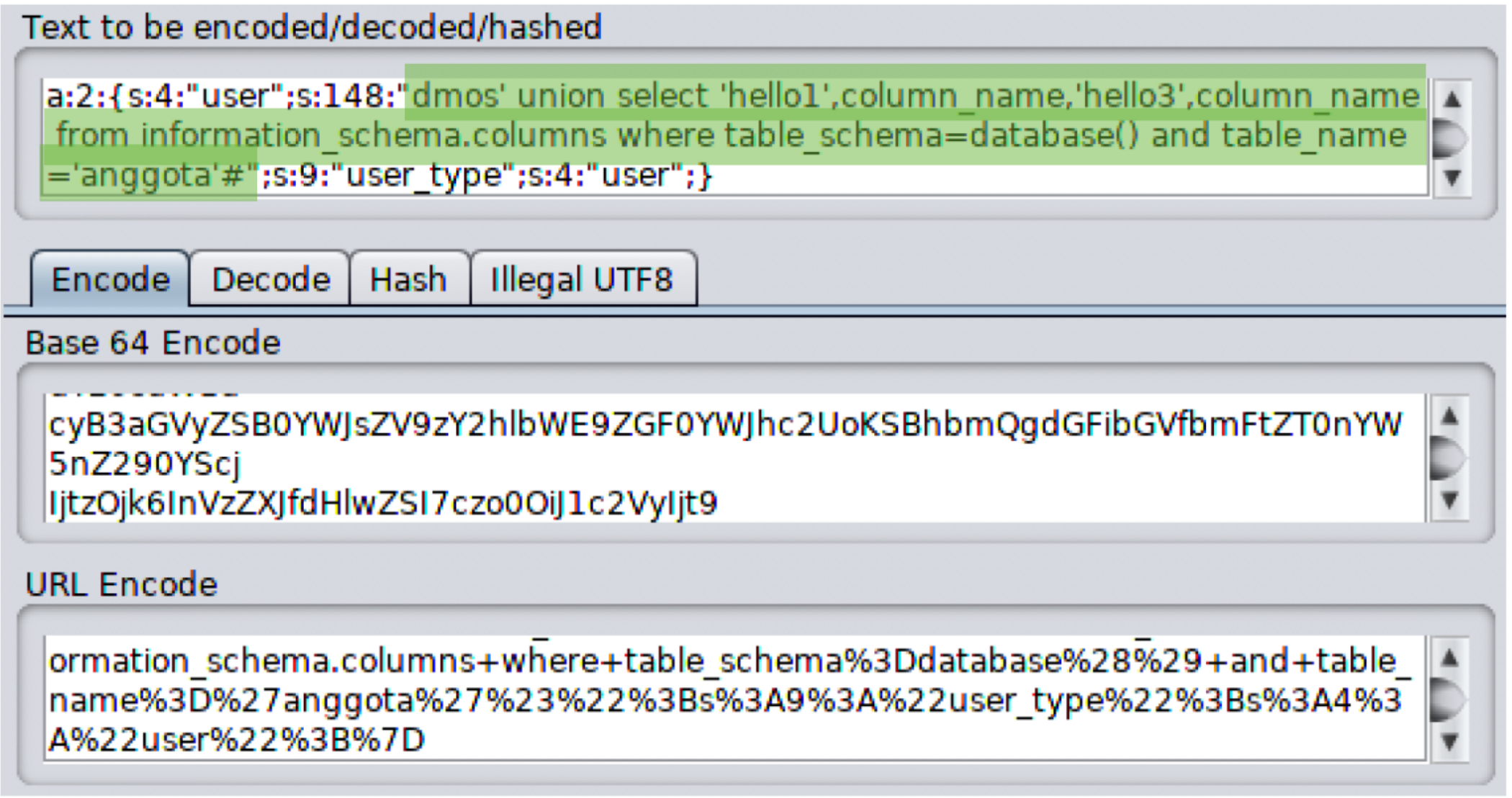


Figure 4.17: Discovery column name

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Microsoft-IIS/7.5
X-Allocated-On-The-Fly-By: Coliseum Web Application Security Framework
X-Powered-By: Caendra Team
Date: Fri, 28 Jun 2019 10:55:44 GMT
Content-Length: 172
```

```
login successfuly<br><br>Hi <b>dmos</b><h1>Sorry, You are just an user :( !<
/h1>Hi <b>id</b><h1>Sorry, You are just an id :( !</h1><a href="index.php?
act=logout">Logout</a>
```

Figure 4.18: Response

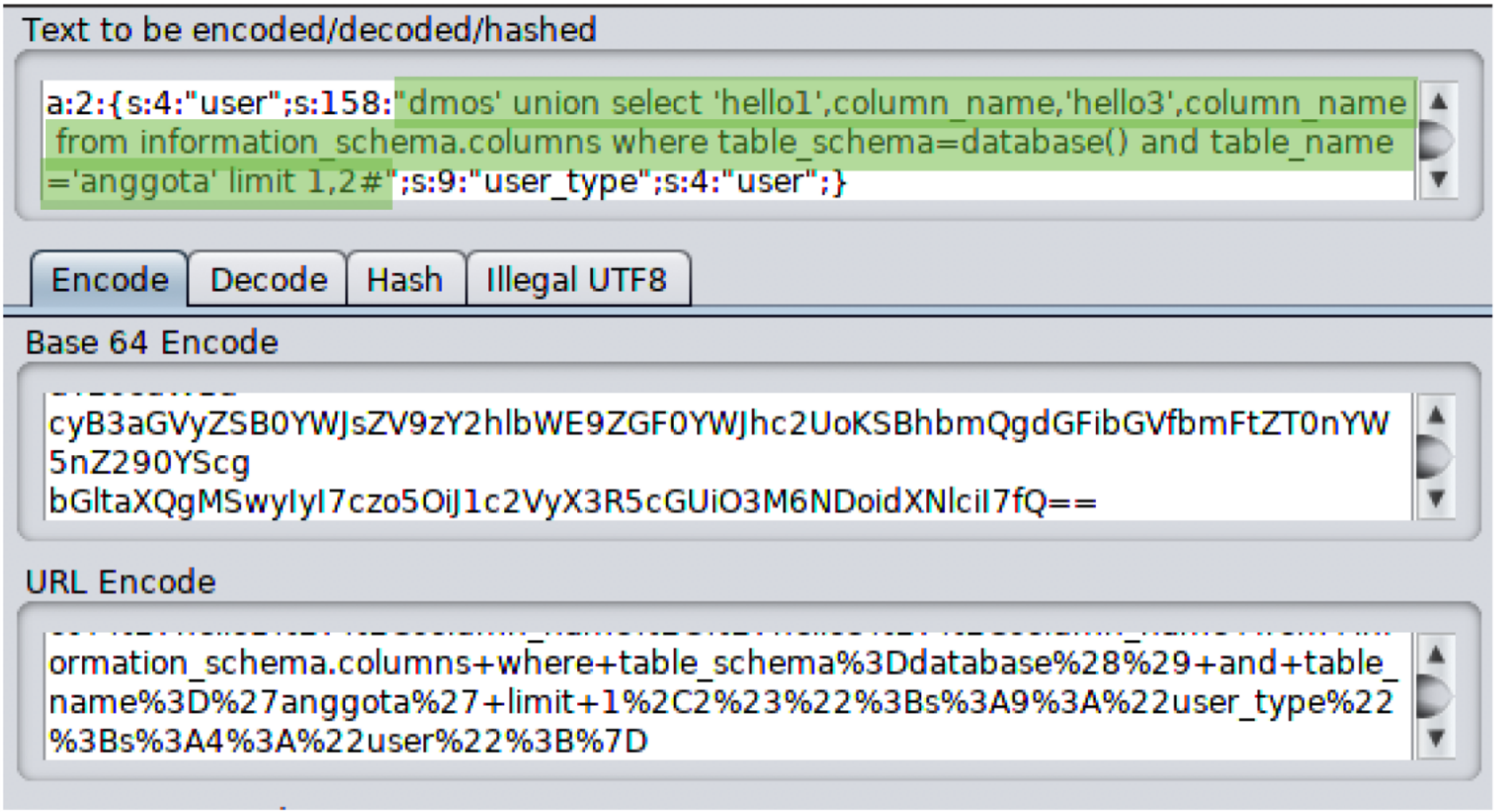


Figure 4.19: Discovery column name

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Microsoft-IIS/7.5
X-Allocated-On-The-Fly-By: Coliseum Web Application Security Framework
X-Powered-By: Caendra Team
Date: Fri, 28 Jun 2019 10:58:29 GMT
Content-Length: 172
```

```
login successfuly<br><br>Hi <b>id</b><h1>Sorry, You are just an id :( !</h1>
Hi <b>user</b><h1>Sorry, You are just an user :( !</h1><a href="index.php?
act=logout">Logout</a>
```

Figure 4.20: Response

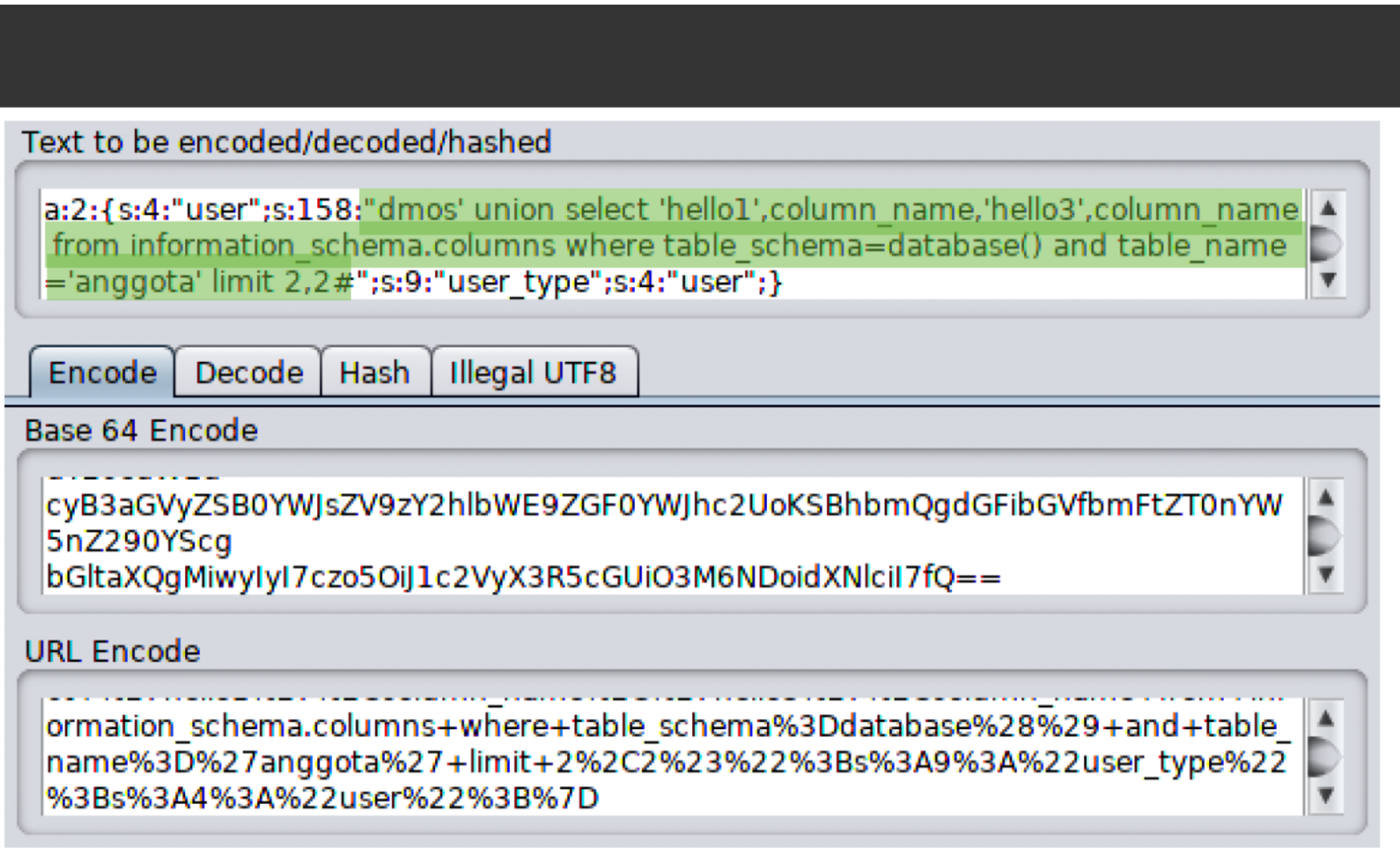


Figure 4.21: Discovery column name

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Microsoft-IIS/7.5
X-Allocated-On-The-Fly-By: Coliseum Web Application Security Framework
X-Powered-By: Caendra Team
Date: Fri, 28 Jun 2019 11:00:23 GMT
Content-Length: 184
```

```
login successfuly<br><br>Hi <b>user</b><h1>Sorry, You are just an user :( !<
/h1>Hi <b>password</b><h1>Sorry, You are just an password :( !</h1><a href="
index.php?act=logout">Logout</a>
```

Figure 4.22: Response

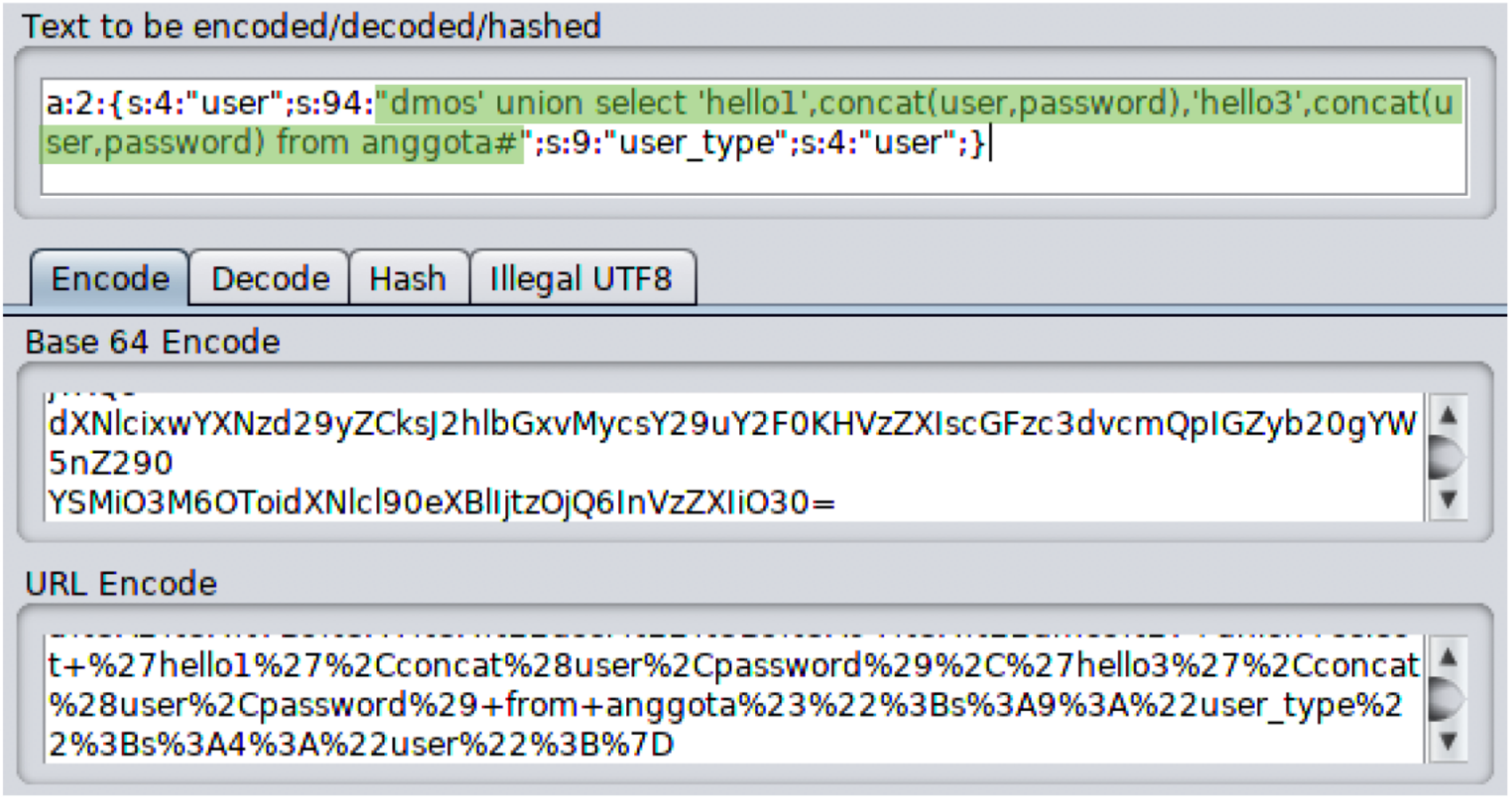


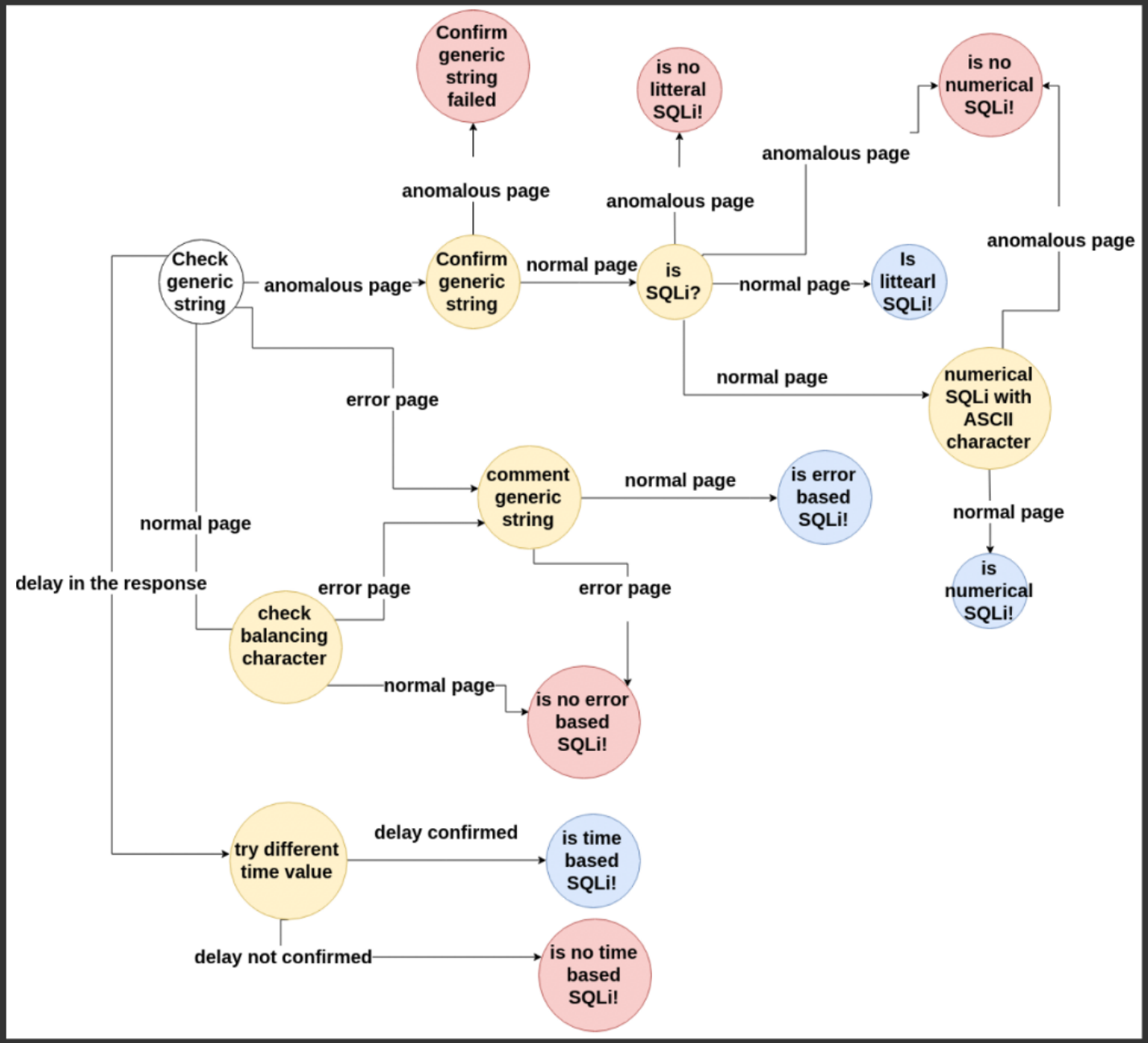
Figure 4.23: Data extraction

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Microsoft-IIS/7.5
X-Allocated-On-The-Fly-By: Coliseum Web Application Security Framework
X-Powered-By: Caendra Team
Date: Fri, 28 Jun 2019 11:03:19 GMT
Content-Length: 226
```

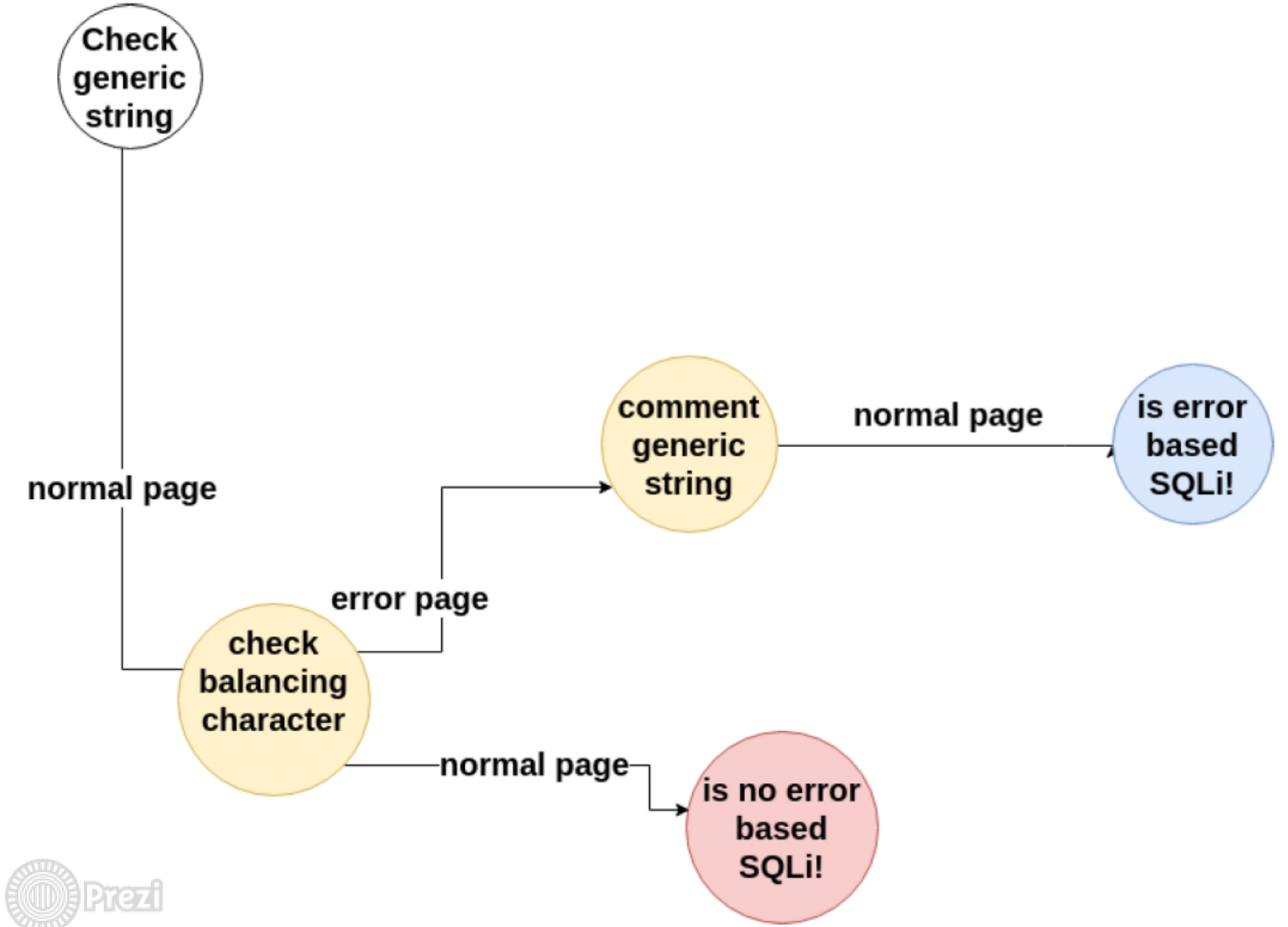
```
login successfuly<br><br>Hi <b>dmos</b><h1>Sorry, You are just an user :( !<
/h1>Hi <b>admincoolczo4OiJQQDU1dzByZCI7</b><h1>Sorry, You are just an
admincoolczo4OiJQQDU1dzByZCI7 :( !</h1><a href="index.php?act=logout">Logout
</a>
```

Figure 4.24: Exploitation success

Un modello di detection per SQL Injection



Prezi


Check generic string
normal page
comment generic string
normal page
is error based SQLi!
error page
check balancing character
normal page
is no error based SQLi!

L'aspetto della generalizzazione

- Una questione di cruciale importanza
- Il mantenimento e l'aggiornamento dei modelli comportamentali si dimostra difficoltoso a causa della continua evoluzione delle tecnologie e delle metodologie di attacco
- Un possibile approccio:
  - impiego di tecniche di *Reinforcement Learning*

generalizzazione

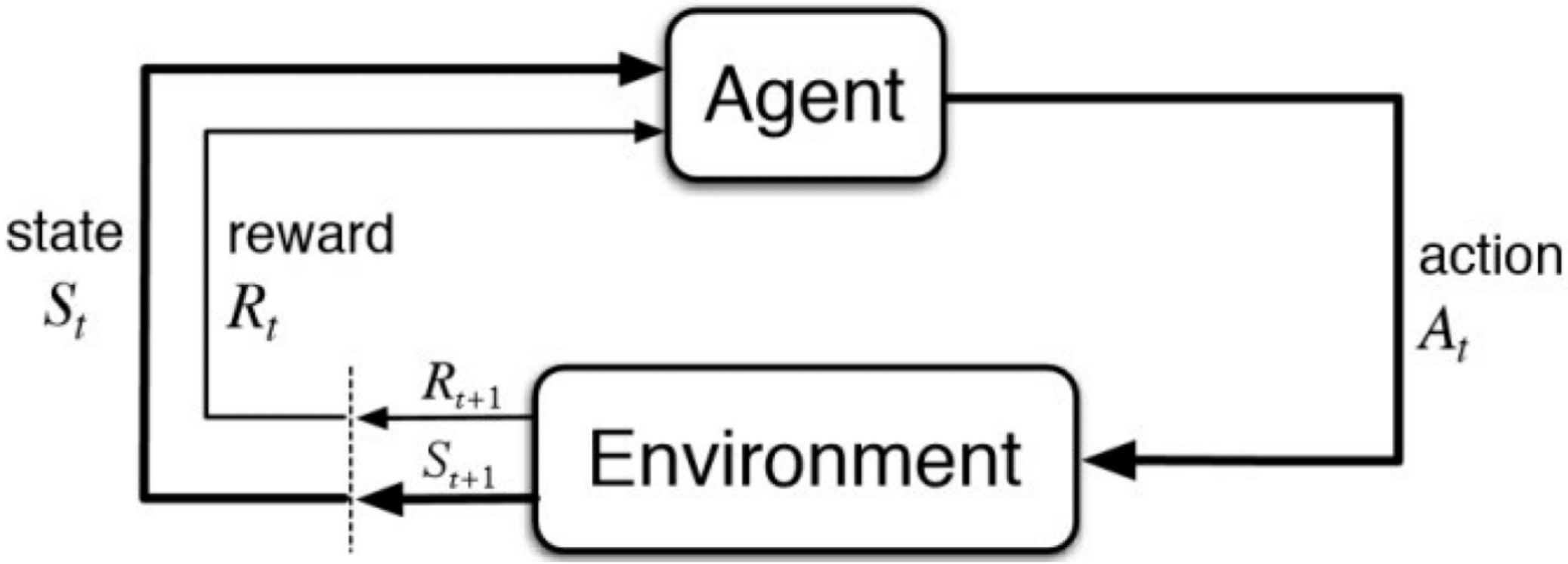
- Una questione di cruciale importanza
- Il mantenimento e l'aggiornamento dei modelli comportamentali si dimostra difficoltoso a causa della continua evoluzione delle tecnologie e delle metodologie di attacco
- Un possibile approccio:
    - impiego di tecniche di *Reinforcement Learning*

Reinforcement Learning

- Prevede che un agente selezioni in ogni istante di tempo un'azione da eseguire nell'ambiente che si intende analizzare
- L'ambiente restituirà all'agente una osservazione, tipicamente formalizzata in termini di:
  - ricompensa (reward) per l'azione svolta
  - relativo stato prossimo
- Senza alcun tipo di addestramento, l'agente produce un output in termini di insieme delle azioni "migliori" (quelle che hanno massimizzato il reward)

Learning

- Prevede che un agente selezioni in ogni istante di tempo un'azione da eseguire nell'ambiente che si intende analizzare
- L'ambiente restituirà all'agente una osservazione, tipicamente formalizzata in termini di:
  - ricompensa (reward) per l'azione svolta
  - relativo stato prossimo
- Senza alcun tipo di addestramento, l'agente produce un output in termini di insieme delle azioni "migliori" (quelle che hanno massimizzato il reward)



Un esempio: Q-learning

- Ottimizzazione delle decisioni in presenza di processi decisionali di tipo markoviano
- Ad ogni istante di tempo, l'agente osserva lo stato del sistema ed applica una determinata azione
- Il processo si sposta in un nuovo stato e l'agente riceve un "reward" che dipende sia dallo stato precedente, che dall'azione in esso compiuta
- L'obiettivo dell'apprendimento è quello di trovare l'ordine sequenziale di azioni che massimizza la somma dei reward futuri:
  - in pratica, il percorso minimo tra stato iniziale e stato finale

Un esempio: Q-learning

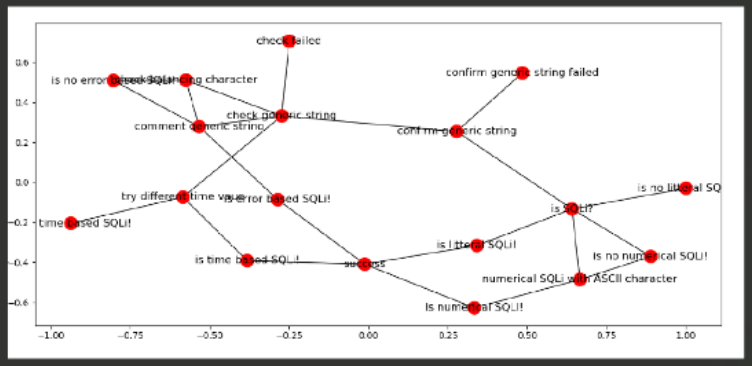
- Ottimizzazione delle decisioni in presenza di processi decisionali di tipo markoviano
- Ad ogni istante di tempo, l'agente osserva lo stato del sistema ed applica una determinata azione
- Il processo si sposta in un nuovo stato e l'agente riceve un "reward" che dipende sia dallo stato precedente, che dall'azione in esso compiuta
- L'obiettivo dell'apprendimento è quello di trovare l'ordine sequenziale di azioni che massimizza la somma dei reward futuri:
    - in pratica, il percorso minimo tra stato iniziale e stato finale

- Il valore gamma indica una sorta di "fattore di sconto", nel range [0,1]:
  - valori prossimi allo zero consentono all'agente di considerare solo il reward relativo allo stato corrente
  - valori prossimi ad uno renderanno l'agente più attento alle possibili ricompense future

$$Q(S_t, A_t) = R(S_t, A_t) + \gamma * Max[Q(S_{t+1}, A_t)]$$

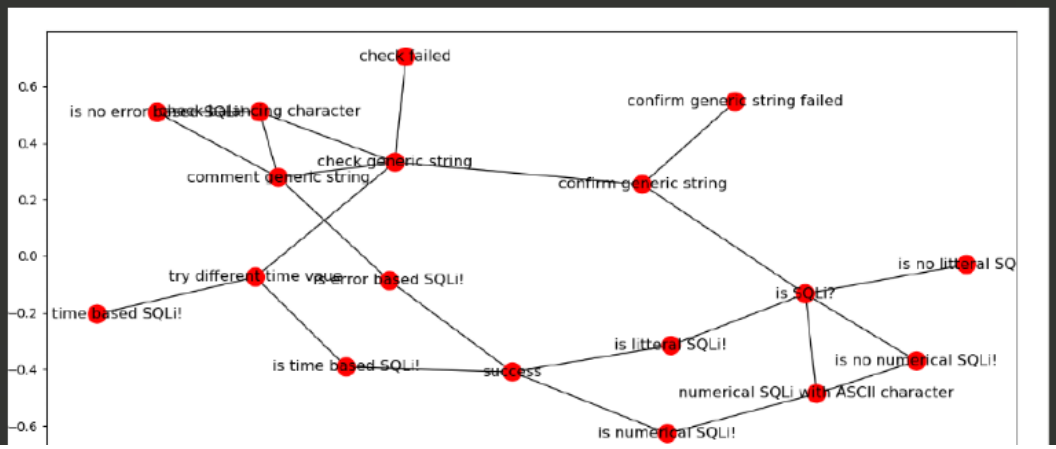
Q-learning ed SQL injection

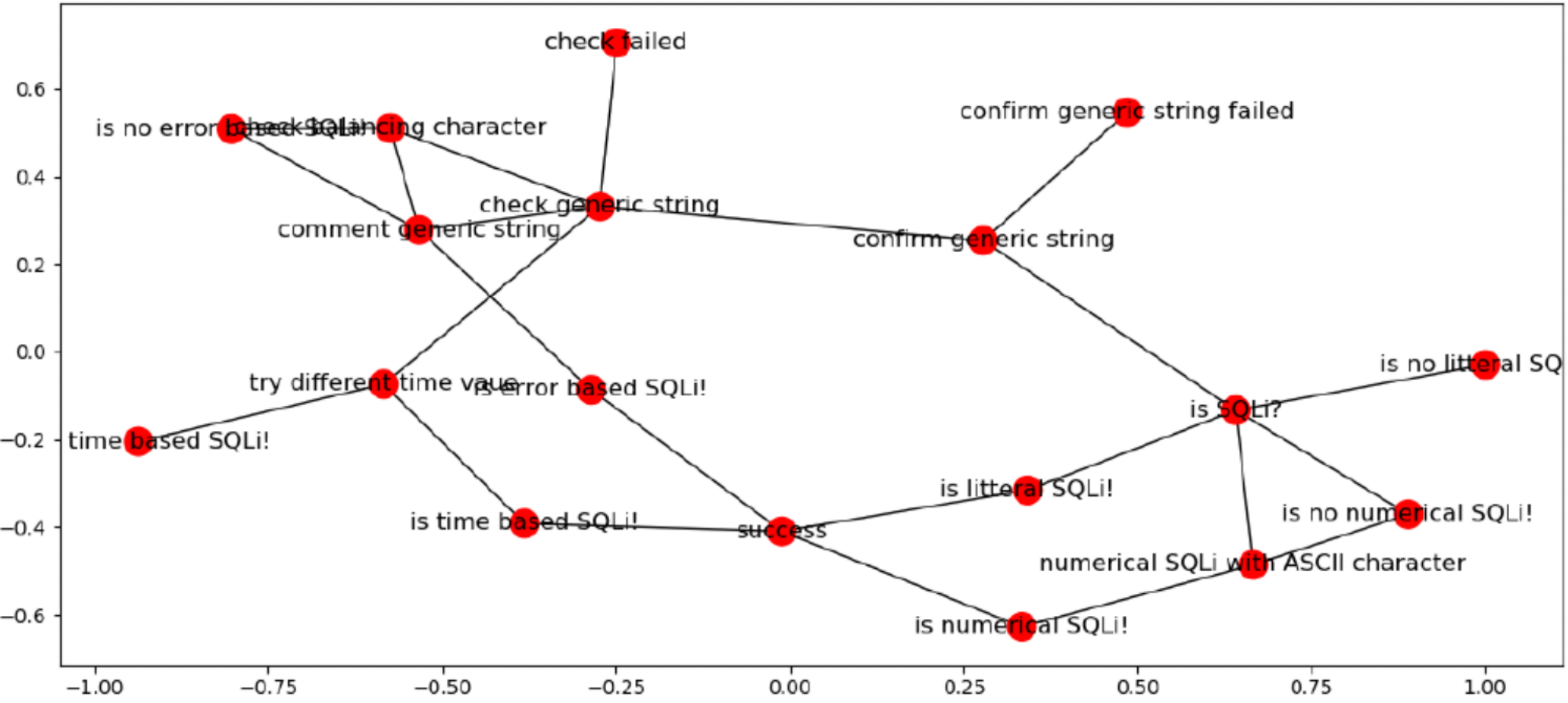
- L'agente interagisce con l'ambiente effettuando azioni ed osservando le loro conseguenze (stati successivi)
- L'ambiente definisce, dunque, sia gli stati che le ricompense



Q-learning ed SQL injection

- L'agente interagisce con l'ambiente effettuando azioni ed osservando le loro conseguenze (stati successivi)
- L'ambiente definisce, dunque, sia gli stati che le ricompense




check failed
is no error based SQLi
balancing character
confirm generic string failed
check generic string
comment generic string
confirm generic string
try different time value
is error based SQLi!
is no litteral SQ
is SQLi?
time based SQLi!
is litteral SQLi!
is time based SQLi!
success
is no numerical SQLi!
numerical SQLi with ASCII character
is numerical SQLi!
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

A proposito di reward

- Stati che si trovano lungo un percorso che porta al successo:
  - reward >= 0
- Altri stati:
  - reward < 0
- La somma dei reward è costante lungo ciascun percorso di successo

- Stati che si trovano lungo un percorso che porta al successo:
  - reward >= 0
- Altri stati:
  - reward < 0
- La somma dei reward è costante lungo ciascun percorso di successo

action/state	reward
apex/confirm generic string	0
apex/comment generic string	2
apex/check balancing character	0
apex/try different time value	0
double apex/is SQLi?	0
apex/try different time value	0
input value as concat function/is litteral SQLi!	5
input value as math expression/numerical SQLi with ASCII character	0
input value as math expression with ASCII character/is numerical SQLi!	5
comment string/is error based SQLi!	3
balancing string/comment generic string	2
differnt time value/is time based SQLi!	5

Table 4.1: Tabella dei reward

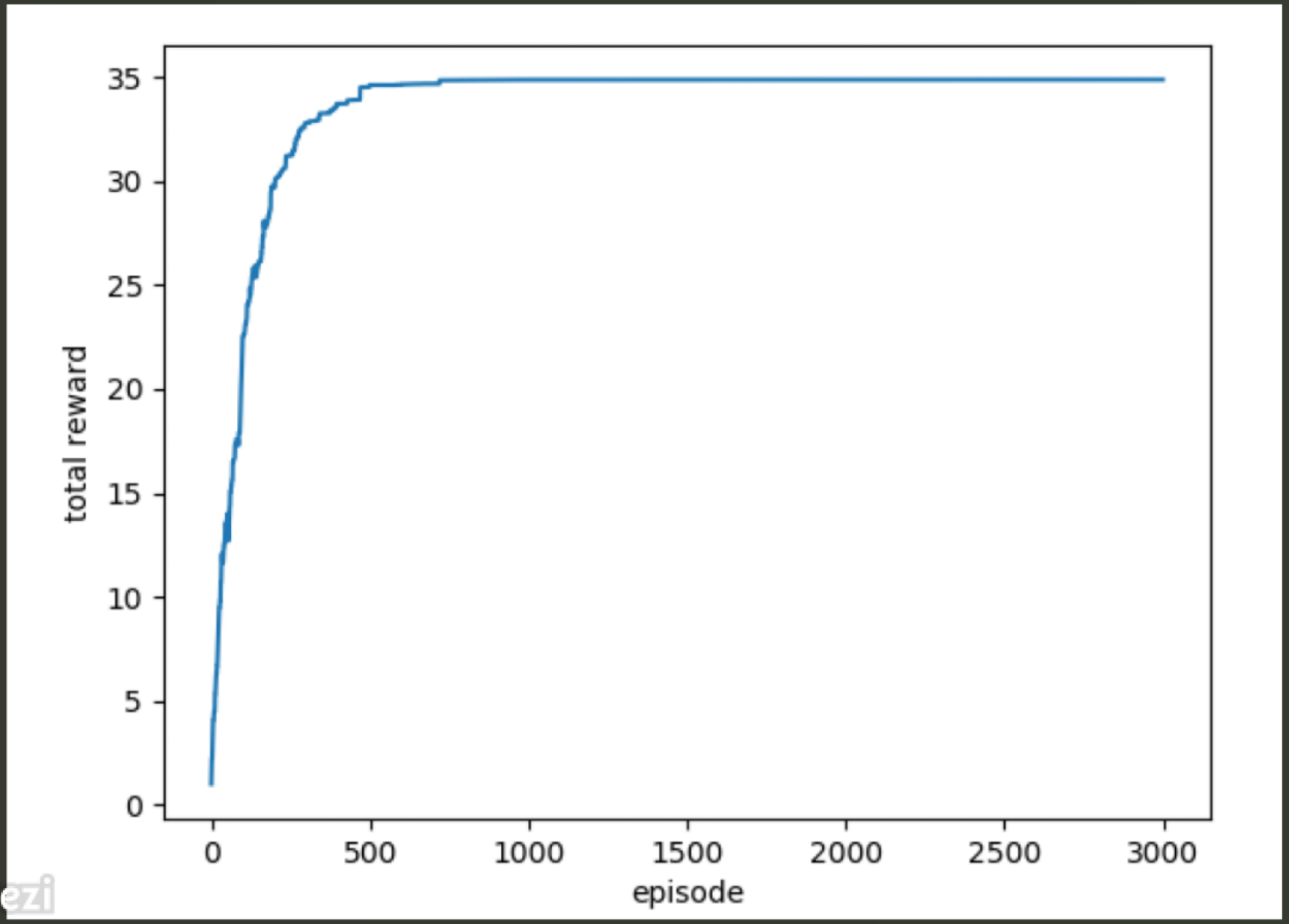
Testing effettuato su una challenge messa a disposizione dal sito hack.me

https://hack.me/101522/sql-injection-medium.html

L'agente aggiorna la tabella ad ogni iterazione...

0.	0.76	0.85	0.	0.	0.	0.	0.	0.	0.	1.	0.	1.	0.	0.95	0.	0.	0.
0.6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.6	0.	0.	0.6	0.75	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.45	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.45	0.	0.	0.76	0.76	1.	0.99	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.6	0.	0.	0.	0.6	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.
0.	0.	0.	0.	0.6	0.	0.76	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.6	0.	0.	0.	0.	0.	0.	0.	0.	1.
0.6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.99	0.6	0.76	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.59	0.	0.	0.	0.	0.	0.	1.
0.59	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.	0.	0.76	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.59	0.	0.6	0.	0.	0.	0.	0.
0.6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.92	0.73	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.55	0.	0.	1.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.55	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.6	0.	0.59	0.	0.59	0.	0.	0.	0.59	0.	0.99

- Fattore di sconto (gamma): 0,7
- L'algoritmo converge dopo circa 700 episodi



- Il modello restituisce la sequenza di azioni ottimali per rilevare la vulnerabilità e classificarla come "error-based SQLi":
  - apice ('), seguito da commento (#)

```
Most efficient path:
[0, 10, 11, 17]
'
#
--
--+-
--+
dmos@dmos-Lenovo-G50-80:~/Scrivania/codiceRL$
```

Conclusioni

- Un approccio innovativo al rilevamento di vulnerabilità di tipo SQL Injection
- Caratterizzazione e modellizzazione delle dinamiche comportamentali di un Penetration Tester
- Riproduzione automatizzata di tali dinamiche allo scopo di velocizzare le attività di vulnerability assessment
- Studio preliminare dell'efficacia derivante dalla adozione di approcci basati sul Reinforcement Learning

Conclusioni

- Un approccio innovativo al rilevamento di vulnerabilità di tipo SQL Injection
- Caratterizzazione e modellizzazione delle dinamiche comportamentali di un Penetration Tester
- Riproduzione automatizzata di tali dinamiche allo scopo di velocizzare le attività di vulnerability assessment
- Studio preliminare dell'efficacia derivante dalla adozione di approcci basati sul Reinforcement Learning

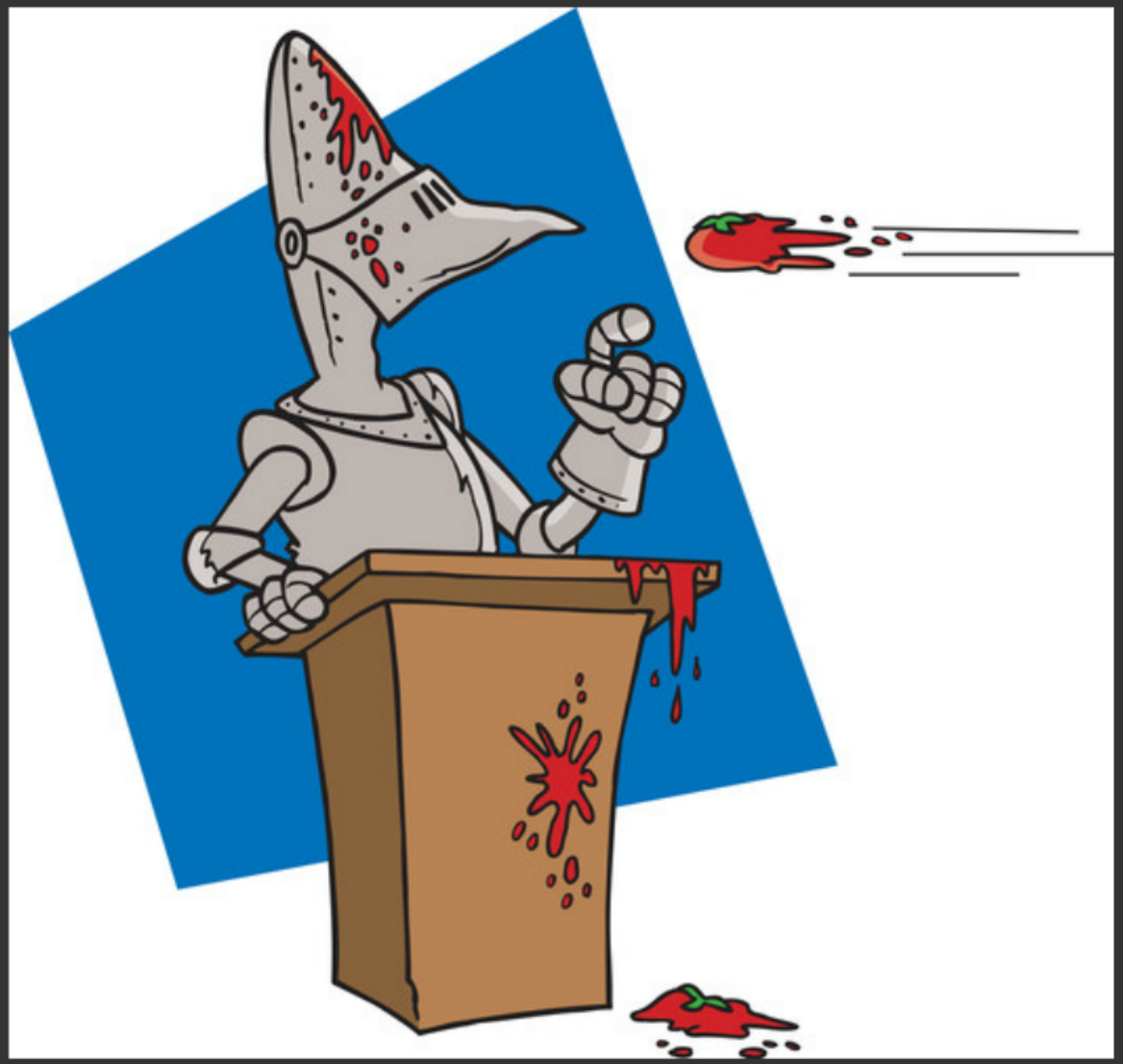
Direzioni di sviluppo

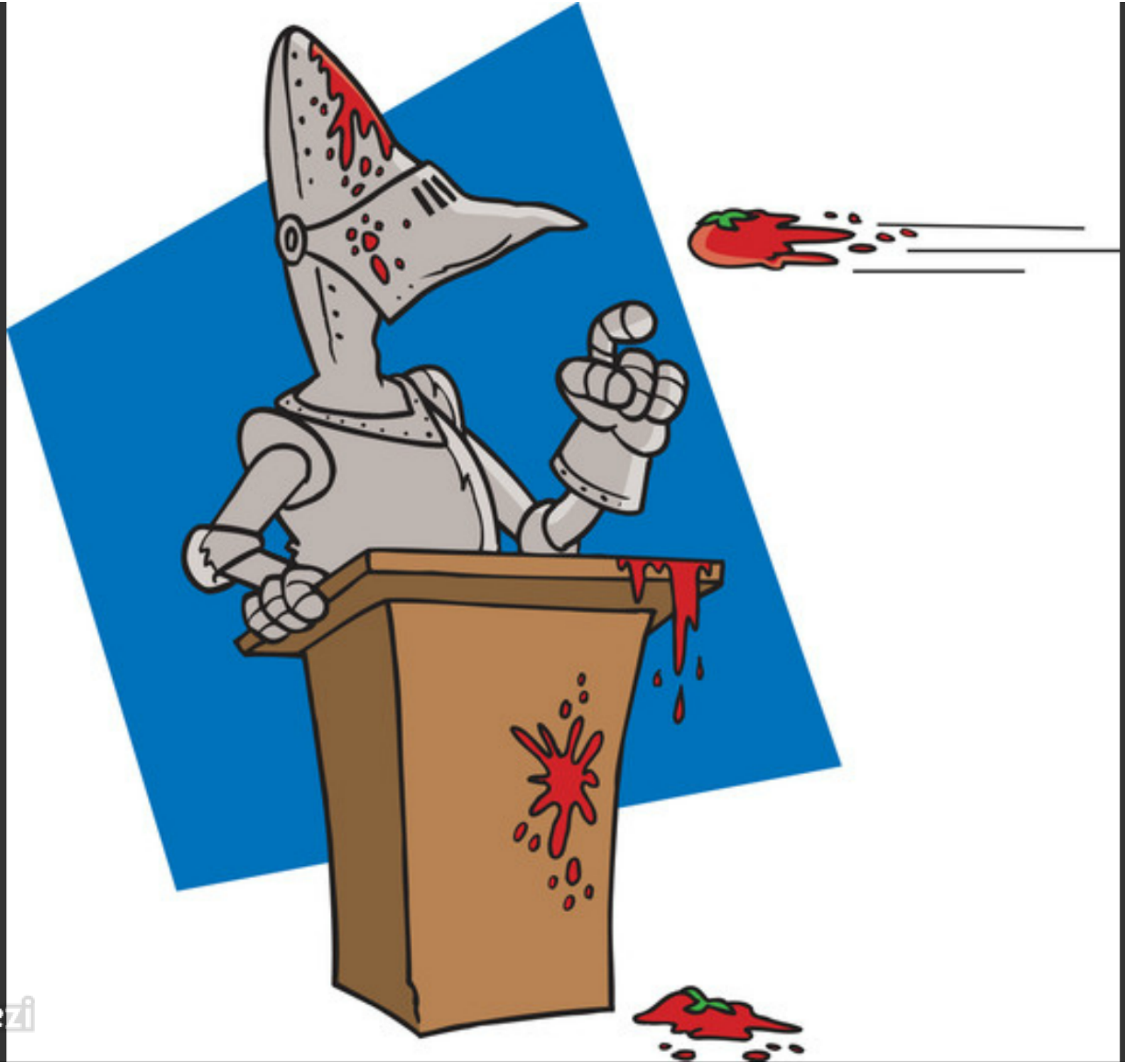
- Applicare il medesimo approccio anche alla fase di exploitation
- Considerare altri tipi di vulnerabilità:
  - lavoro già svolto in ambito XSS (Cross Site Scripting
  - risultati seminali già ottenuti nell'ambito del cosiddetto "Generic Testing" (quindi non solo web applications...)
- Analizzare l'efficacia di eventuali approcci alternativi
- Proseguire le attività di trasferimento tecnologico:
  - collaborazione attiva, in questo ambito, con NTT Data (Business Units di Security e di Intelligenza Artificiale)

Direzioni di sviluppo

- Applicare il medesimo approccio anche alla fase di exploitation
- Considerare altri tipi di vulnerabilità:
  - lavoro già svolto in ambito XSS (Cross Site Scripting
  - risultati seminali già ottenuti nell'ambito del cosiddetto "Generic Testing" (quindi non solo web applications...)
- Analizzare l'efficacia di eventuali approcci alternativi
- Proseguire le attività di trasferimento tecnologico:
  - collaborazione attiva, in questo ambito, con NTT Data (Business Units di Security e di Intelligenza Artificiale)

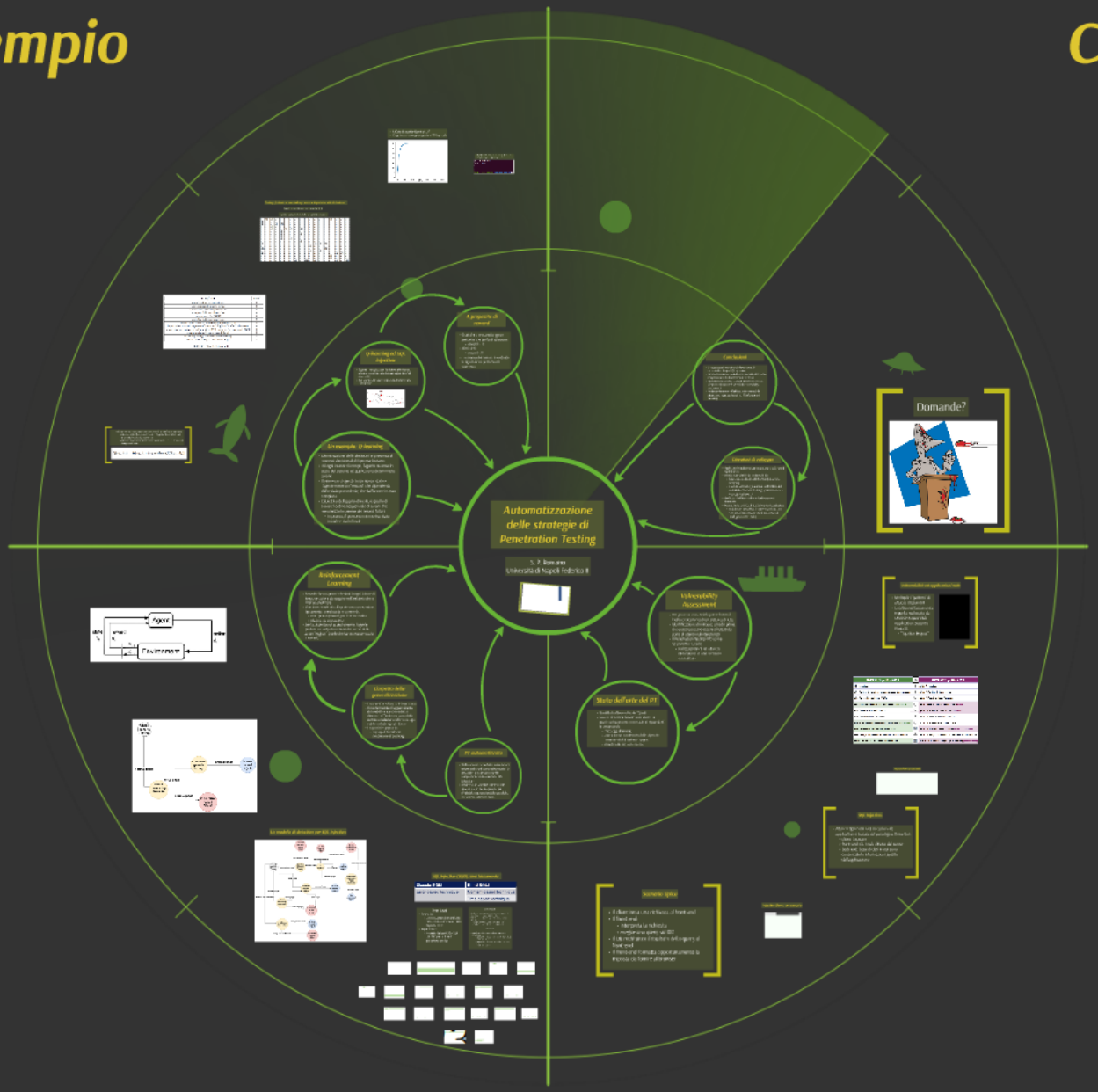
Domande?





Caso di esempio

Conclusioni



Contributo

Contesto

WORKSHOP GARR 2019
NET MAKERS
LA COMUNITÀ CHE INNOVA LA RETE

Prezi