TÉCNICO LISBOA

Segurança em Software

Mestrado em Engenharia Informática e de Computadores

Projeto de Segurança em Software

–

Discovering vulnerabilities in PHP web applications

Prof. Ana Matos
Prof. Pedro Adão

Grupo 15

Pedro Lopes – 81988
Bruno Santos – 82053
Afonso Caetano – 82539

Introdução

No âmbito da cadeira de Segurança em Software foi-nos proposto que desenvolvêssemos um projeto com o intuito de estudar como as vulnerabilidades em código PHP podem ser detetadas estaticamente em termos de validação de input.

De acordo com o *OWASP's Top Ten Project*, documento que lista as dez fontes mais relevantes de vulnerabilidades em aplicações web, a maior causa de vulnerabilidades em aplicações web ocorrem por meio de input não validado.

Desenho da ferramenta

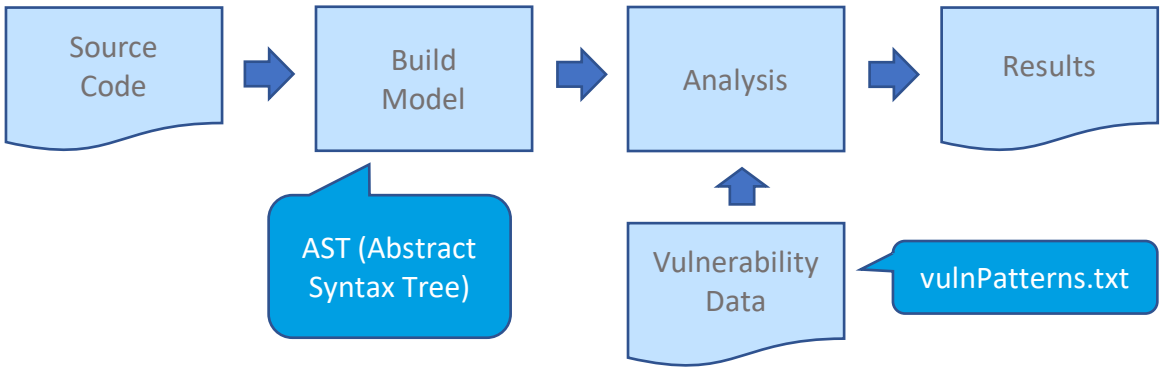
Uma vez que as ferramentas de análise estática são complexas, a ferramenta desenvolvida foi projetada para analisar *slices* de programas PHP. Entenda-se uma *slice* como a sequência de instruções que podem afetar o fluxo de dados desde um ponto de entrada (*entry point*) até um *sensitive sink*.

Escolhemos desenvolver a ferramenta em linguagem Python devido aos seguintes fatores:

- Suporte extenso de diversas bibliotecas, sendo que neste caso foram utilizadas as bibliotecas "json" e "sys";
- É uma linguagem de alto nível, aproximando-se assim mais da linguagem humana abstraindo de pormenores da arquitetura da máquina;
- Suporte para programação funcional

A nossa ferramenta está preparada para encontrar todas as vulnerabilidades enunciadas na tabela[1] através de análise estática onde a nível de precisão

[1] http://awap.sourceforge.net/support.html



Exemplos

De seguida, mostramos dois exemplos de código PHP analisado pelo nosso programa e o respetivo output:

- Este código PHP é vulnerável a ataques do tipo SQL Injection:

```
<?php
$nis=$_POST['nis'];
$query="SELECT *FROM siswa WHERE nis='$nis'";
$q=mysql_query($query,$koneksi);
?>
```

O output gerado pela ferramenta é o seguinte:

```
> Program is vulnerable!
> Type of vulnerability: SQL injection
> Possible correction(s):
mysql_escape_string
mysql_real_escape_string
```

- Este código PHP não é vulnerável porque tem uma função de sanitização (bem aplicada):

```
<?php
$nis=$_POST['nis'];
$query="SELECT *FROM siswa WHERE nis='$nis'";
$query=mysql_real_escape_string($query);
$q=mysql_query($query,$koneksi);
?>
```

O output gerado pela ferramenta é o seguinte:

```
> Program is not vulnerable!
> Due to: mysql_escape_string
```

NOTA: due to: mysql_real_escape_string

A frase "o output gerado ..." não devia tar antes de mostrar o output em si?

O output foi alterado para que as correçoes possiveis apareçam na mesma linha em vez de uma por linha

Discussão

Dadas as limitações intrínsecas associadas à analise estática (“”), a ferramenta desenvolvida é necessariamente imprecisa, podendo ser não confiável (produz falsos negativos), incompleto (produzir falsos positivos) ou mesmo ambos. Assim, de seguida, será analisada a imprecisão da ferramenta desenvolvida, a maneira como esta pode ser minimizada e, por último, uma proposta de melhoramento da ferramenta, de maneira a torna-la mais precisa.