The research on vulnerability analysis in OpenADR for Smart Grid

Korea University
Formal Methods Lab
Mijeong Park
mjpark@formal.korea.ac.kr

Introduction – SW Assurance (SwA)

- **SW Assurance**
  - > **the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at anytime during its lifecycle, and that the software functions in the intended manner." – CNSS IA Glossary, 2010**

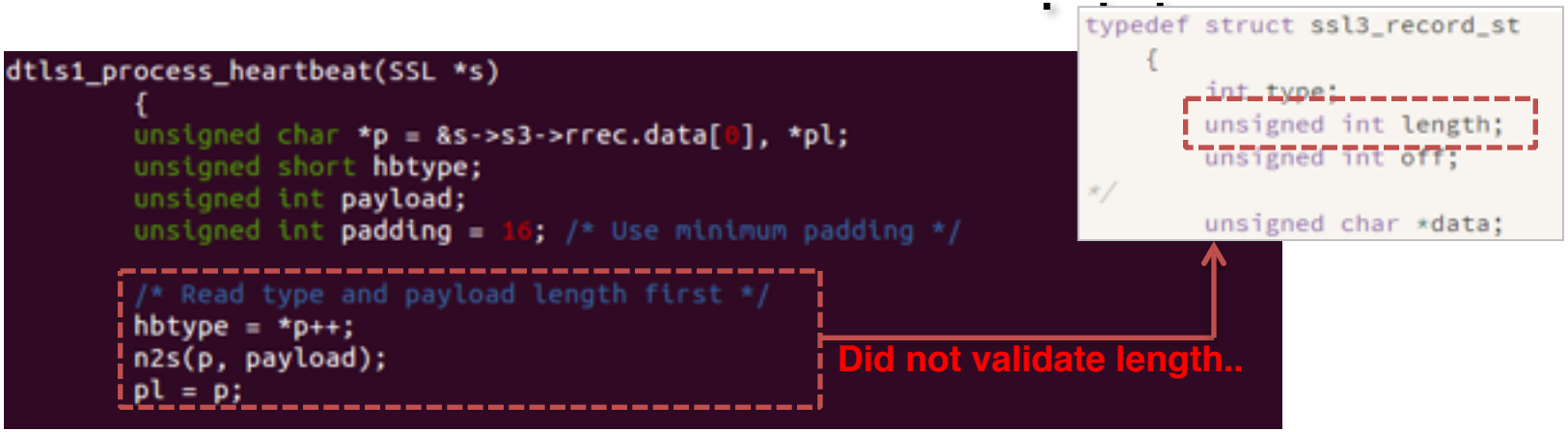
SW Quality Assurance

+

SW Security Assurance

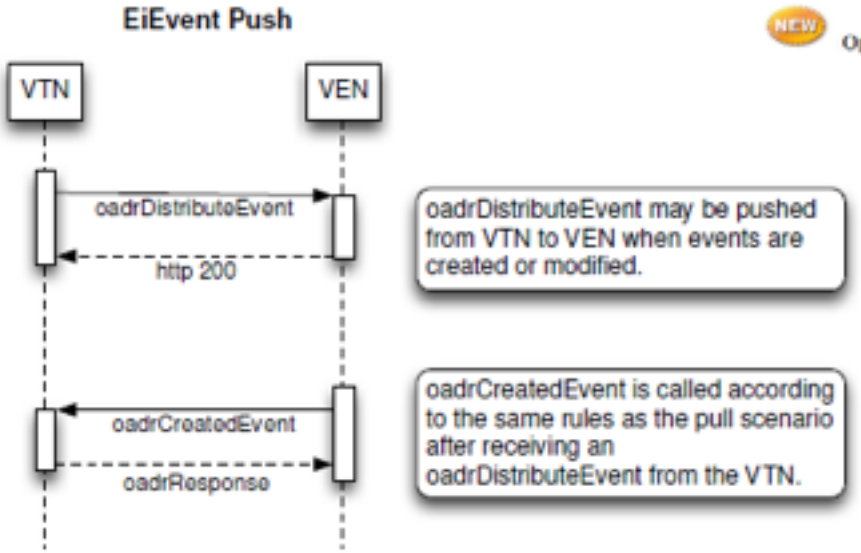
Introduction – The crisis of Open Source SW

- **OpenSSL HeartBleed**
  - > **CVE-2014-0160**
    - : The (1) TLS and (2) DTLS implementations in OpenSSL 1.0.1 before 1.0.1g do not properly handle Heartbeat Extension packets, which allows remote attackers to obtain sensitive information from process memory …
  - > **OpenSSL 1.0.1 ~ OpenSSL 1.0.1f**
  - > **in Heartbeat Extension**



Related works - OpenADR

- OpenADR(Open Automated Demand Response)
- Smart Grid Protocol for Automated DR(Demand Response)
- Overseas: The effect reducing the load of electric power
- Domestic: The plan to adopt OpenADR as the standard



Analysis – The result of LDRA

- **The result of LDRA Testbed based on CERT Java**

CERT code	Number of Violations (VEN)	Number of Violations (VTN)	Total
OBJ01-J	627	93	**720**
OBJ10-J	404	28	**432**
MET02-J	0	0	**0**
ERR03-J	2	3	**5**
THI00-J	0	0	**0**
THI05-J	0	0	**0**
FIO02-J	0	0	**0**
FIO04-J	0	0	**0**
FIO09-J	0	0	**0**
MSC01-J	7	7	**14**
MSC02-J	0	0	**0**

CERT code	Number of Violations (VEN)	Number of Violations (VTN)	Total
IDS05-J	0	0	**0**
IDS06-J	0	0	**0**
EXP00-J	0	2	**2**
EXP05-J	130	5	**135**
EXP06-J	0	0	**0**
NUM00-J	0	0	**0**
NUM01-J	0	0	**0**
NUM02-J	0	0	**0**
NUM07-J	0	0	**0**
NUM09-J	0	0	**0**
NUM12-J	0	5	**5**
NUM13-J	0	0	**0**

- **OBJ01-J**
  - > Limit extensibility of classes and methods with invariants
  - > MITRE CWE-766. Critical variable declared public
  - > CVE-2010-3860

- **EXP00-J**
  - > Do not ignore values returned by methods
  - > MITRE CWE-252. Unchecked return value
  - > CVE-2010-0211

Analysis – OBJ01-J

- **Limit extensibility of classes and methods with invariants**

> **MITRE CWE-766**
  : Critical variable declared public

> **CVE-2010-3860**
  : IcedTea 1.7.x before 1.7.6, 1.8.x before 1.8.3, and 1.9.x before 1.9.2, as based on OpenJDK 6, declares multiple sensitive variables as public, which allows remote attackers to obtain sensitive information including (1) user.name, (2) user.home, and (3) java.home system properties, and other sensitive information such as installation directories.

Violation Example

```
public class ContentType implements Serializable, Equals, HashCode, ToString
{

    protected List<Object> content;
    protected String src;

    private final static long serialVersionUID = 1L;
```

Modification Example

```
public class ContentType implements Serializable, Equals, HashCode, ToString
{

    private List<Object> content;
    private String src;

    public void setSrc(String value) {
        // add codes for validating input value
        this.src = value;
    }
```

Analysis – EXP00-J

- **Do not ignore values returned by methods**

> **MITRE CWE-252**
  : Unchecked Return Value

> **CVE-2010-0211**
  : The slap_modrdn2mods function in modrdn.c in OpenLDAP 2.4.22 does not check the return value of a call to the smr_normalize function, which allows remote attackers to cause a denial of service (segmentation fault) and possibly execute arbitrary code via a modrdn call with an RDN string containing invalid UTF-8 sequences …

Violation Example

```
protected Map<String, Collection<CacheOperationContext>> createOperationContext(
        Collection<CacheOperation> cacheOperations, Method method,
        Class<?> targetClass, HttpServletRequest request) {

    Map<String, Collection<CacheOperationContext>> map = new LinkedHashMap<String, Collection<CacheOperationContext>>(3);

    Collection<CacheOperationContext> cacheables = new ArrayList<CacheOperationContext>();
    Collection<CacheOperationContext> evicts = new ArrayList<CacheOperationContext>();
    Collection<CacheOperationContext> updates = new ArrayList<CacheOperationContext>();

    Object[] args = findArgs(request, method);
```

Modification Example

```
protected Map<String, Collection<CacheOperationContext>> createOperationContext(
        Collection<CacheOperation> cacheOperations, Method method,
        Class<?> targetClass, HttpServletRequest request) {

    Map<String, Collection<CacheOperationContext>> map = new LinkedHashMap<String, Collection<CacheOperationContext>>(3);

    Collection<CacheOperationContext> cacheables = new ArrayList<CacheOperationContext>();
    Collection<CacheOperationContext> evicts = new ArrayList<CacheOperationContext>();
    Collection<CacheOperationContext> updates = new ArrayList<CacheOperationContext>();

    Object[] args = findArgs(request, method);

    if(args != null){
        // add codes after checking return value
    }
```

Analysis – The result of Web Test

… ing

- **IDS00-J**
  - > **Sanitize untrusted data passed across a trust boundary**

- **Grails, Groovy, Java, maven, etc.**

Success

Request

Send!

Response

Failure

Request

Send!

If(!Pattern.matches(“[0-9]+”, val)) …

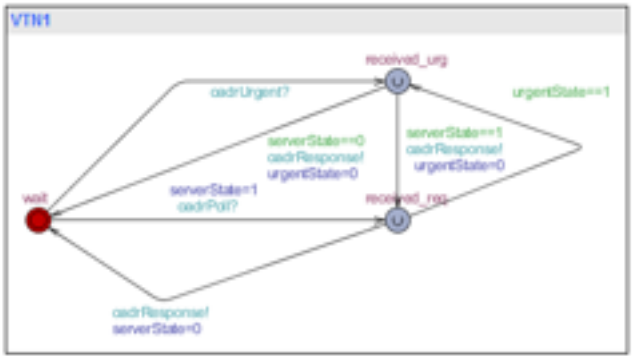
Response

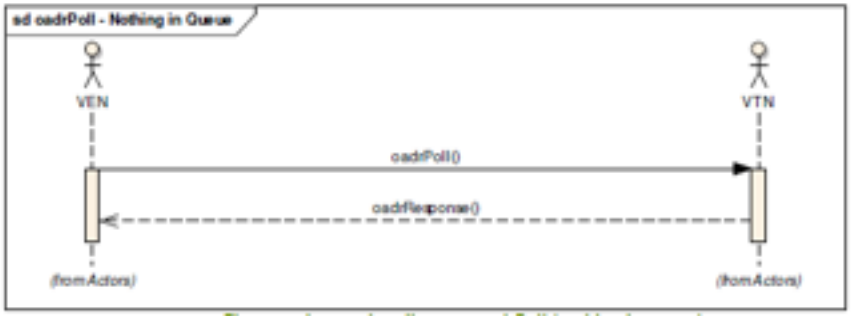
Contribution

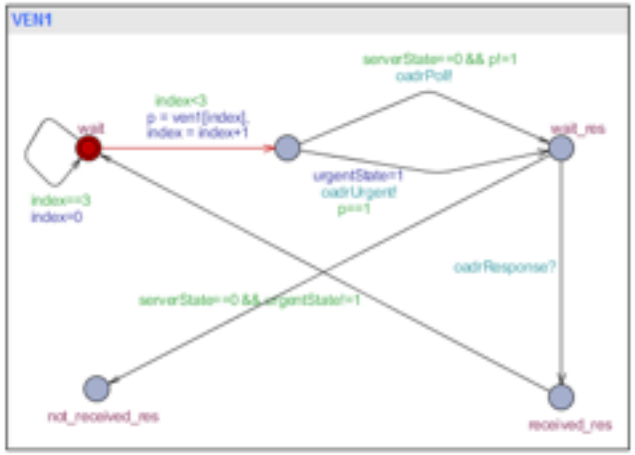
- **SW Assurance**
  - > Quality Assurance + Security Assurance
  - > Using Secure Coding rules for Security Assurance
- **Security Assurance for Open Source SW**
  - > Functionality ↑ Security ↓
- **Smart Grid target**
  - > Open Source of OpenADR

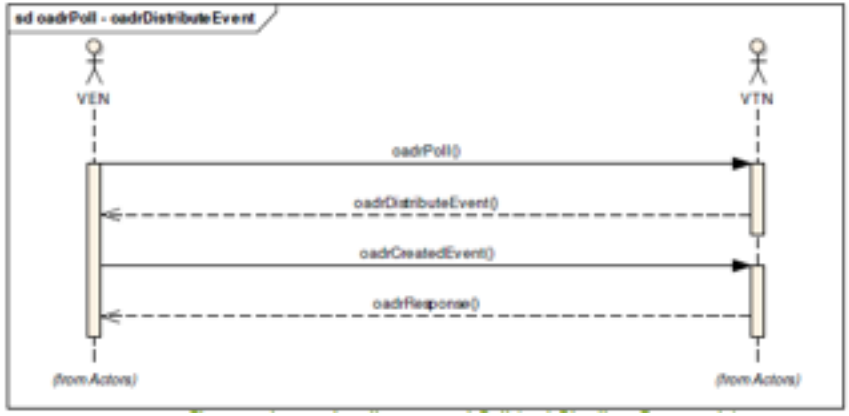
Future Works - Modeling

- **OpenADR Protocol Modeling**
  - > **oadrPoll interaction**
  - > **Using UPPAAL Tool ..**









Thank you

mjpark@formal.korea.ac.kr