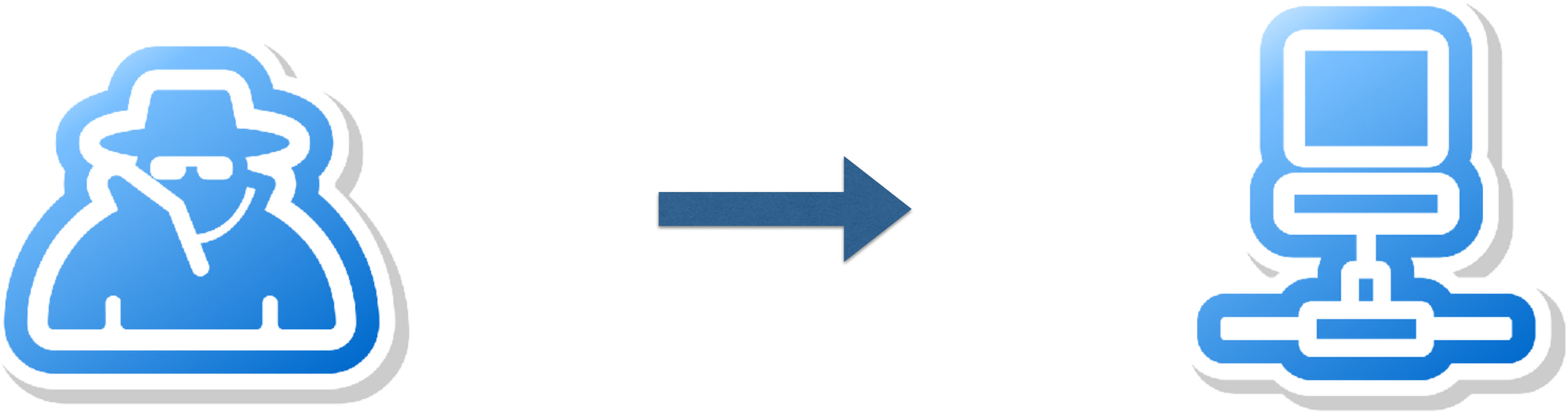
Das Backend schlägt zurück

Safety&Security 2018

Dominik Schadow | bridgingIT



Application Intrusion Detection with OWASP AppSensor

Introduction

Basics

Implementation

Introduction

Attacks are often executed as trial & error

Do you know when your web applications are under attack?

Do you learn about successful attacks before the data has been leaked?

Automatically react on identified attacks in realtime

Stop attackers before they are successful

It’s impossible to detect every attack

It’s enough to know about a devices’s intentions

What makes application intrusion detection more effective than a web application firewall?

?gateway=12345&device=7&temperature=100

Device Manager (Gateway: 12345)

Manage your Devices

1 - Heating
2 - Thermostat
3 - Temperature Detector
4 - Smoke Detector

Manage

?gateway=12345&device=7&temperature=100

Device Manager (Gateway: 12345)

Manage your Devices

1 - Heating
2 - Thermostat
3 - Temperature Detector
4 - Smoke Detector

Manage

?gateway=12345&device=7&temperature=100

Context Matters

Application is aware of business context

Application is aware of valid and invalid data

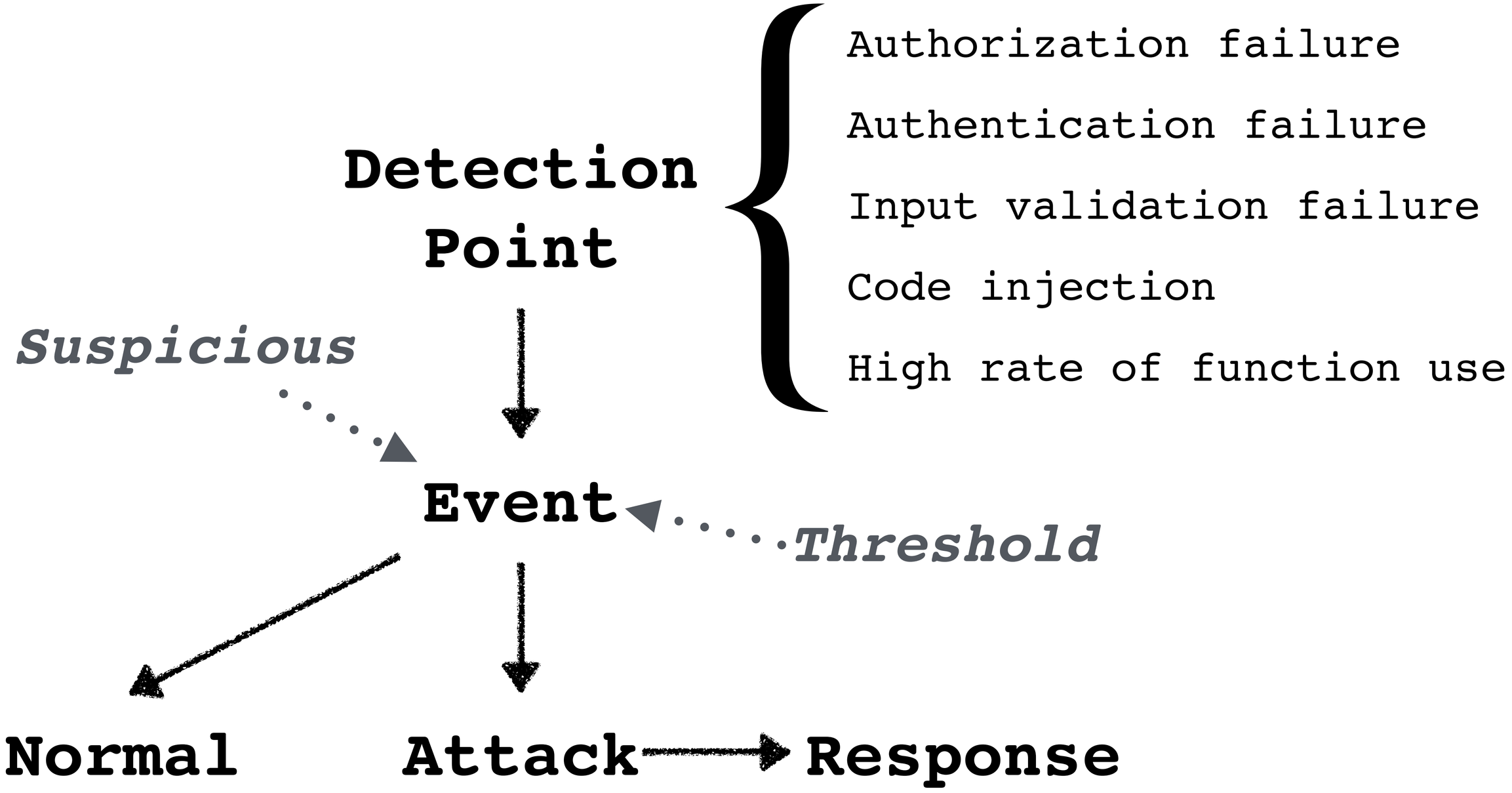
Application aware defense

Part of a defense in depth strategy

Applications and devices have to be secure

Detect attacks, not vulnerabilities

Basics


Detection Point
Authorization failure
Authentication failure
Input validation failure
Code injection
High rate of function use
Suspicious
Event
Threshold
Normal
Attack
Response

Detection Points

Detect events in the backend (like a sensor)

Tightly integrated in the application code

Useable in any backend application layer

```
@Named
public class EncounterValidator implements Validator {
 @Autowired
 private SpringValidatorAdapter validator;

 public void validate(Object target, Errors errors) {
  validator.validate(target, errors);
  // ...
 }
}
```

```
@Named
public class EncounterValidator implements Validator {
 @Autowired
 private SpringValidatorAdapter validator;

 public void validate(Object target, Errors errors) {
  validator.validate(target, errors);
  // ...
 }
}
```

Often already available as isolated notification

```
@Autowired
private DetectionSystem detectionSystem;
@Autowired
private EventManager ids;

public void validate(Object target, Errors errors) {
 validator.validate(target, errors);
 Encounter encounter = (Encounter) target;

 if (hasXssPayload(encounter.getEvent())) {
  fireEvent();
  errors.rejectValue("event", "xss.attempt", "Stop!");
 }
}
```

```
@Autowired
private DetectionSystem detectionSystem;
@Autowired
private EventManager ids;

public void validate(Object target, Errors errors) {
 validator.validate(target, errors);
 Encounter encounter = (Encounter) target;

 if (hasXssPayload(encounter.getEvent())) {
  fireEvent();
  errors.rejectValue("event", "xss.attempt", "Stop!");
 }
}
```

Identification of attacks may require **blacklists**

```
public boolean hasXssPayload(String payload) {
  return containsAnyIgnoreCase(payload, "<", "script",
         "onload", "eval", "document.cookie");
}
```

```
public boolean hasXssPayload(String payload) {
  return containsAnyIgnoreCase(payload, "<", "script",
         "onload", "eval", "document.cookie");
}
```

```
@Autowired
private DetectionSystem detectionSystem;
@Autowired
private EventManager ids;

public void validate(Object target, Errors errors) {
 validator.validate(target, errors);
 Encounter encounter = (Encounter) target;

 if (hasXssPayload(encounter.getEvent())) {
  fireEvent();
  errors.rejectValue("event", "xss.attempt", "Stop!");
 }
}
```

Connect with a central analysis and response unit

Detection Point Categories

Request Exception (RE)

Authentication Exception (AE)

Session Exception (SE)

Access Control Exception (ACE)

Input Exception (IE)

Encoding Exception (EE)

Command Injection Exception (CIE)

FileIO Exception (FIO)

User Trend Exception (UT)

System Trend Exception (STE)

Input Exception

IE1 - Cross-Site Scripting attempt

IE2 - violation of implemented white lists

IE3 - violation of implemented black lists

IE4 - violation of input data integrity

IE5 - violation of stored business data integrity

IE6 - violation of security log integrity

IE7 - detect abnormal content output structure

```
private void fireEvent() {
 DetectionPoint dp = new DetectionPoint(
  DetectionPoint.Category.INPUT_VALIDATION, "IE1");
 ids.addEvent(new Event(user, dp, detectionSystem);
}
```

```
private void fireEvent() {
 DetectionPoint dp = new DetectionPoint(
  DetectionPoint.Category.INPUT_VALIDATION, "IE1");
 ids.addEvent(new Event(user, dp, detectionSystem);
}
```

```
private void fireEvent() {
 DetectionPoint dp = new DetectionPoint(
  DetectionPoint.Category.INPUT_VALIDATION, "IE1");
 ids.addEvent(new Event(user, dp, detectionSystem);
}
```

```
private void fireEvent() {
 DetectionPoint dp = new DetectionPoint(
  DetectionPoint.Category.INPUT_VALIDATION, "IE1");
 ids.addEvent(new Event(user, dp, detectionSystem);
}
```

Events

Observable occurrences in an application

Monitored and analyzed to determine an attack

```
private void fireEvent() {
 DetectionPoint dp = new DetectionPoint(
  DetectionPoint.Category.INPUT_VALIDATION, "IE1");
 ids.addEvent(new Event(user, dp, detectionSystem);
}
```

```
@Autowired
private DetectionSystem detectionSystem;
@Autowired
private EventManager ids;

public void validate(Object target, Errors errors) {
 validator.validate(target, errors);
 Encounter encounter = (Encounter) target;

 if (hasXssPayload(encounter.getEvent())) {
  fireEvent();
  errors.rejectValue("event", "xss.attempt", "Stop!");
 }
}
```

Thresholds

React immediately to malicious events (**threshold = 1**)

Requires **count**

React delayed to suspicious events (**threshold > 1**)

Requires **count** and **interval**

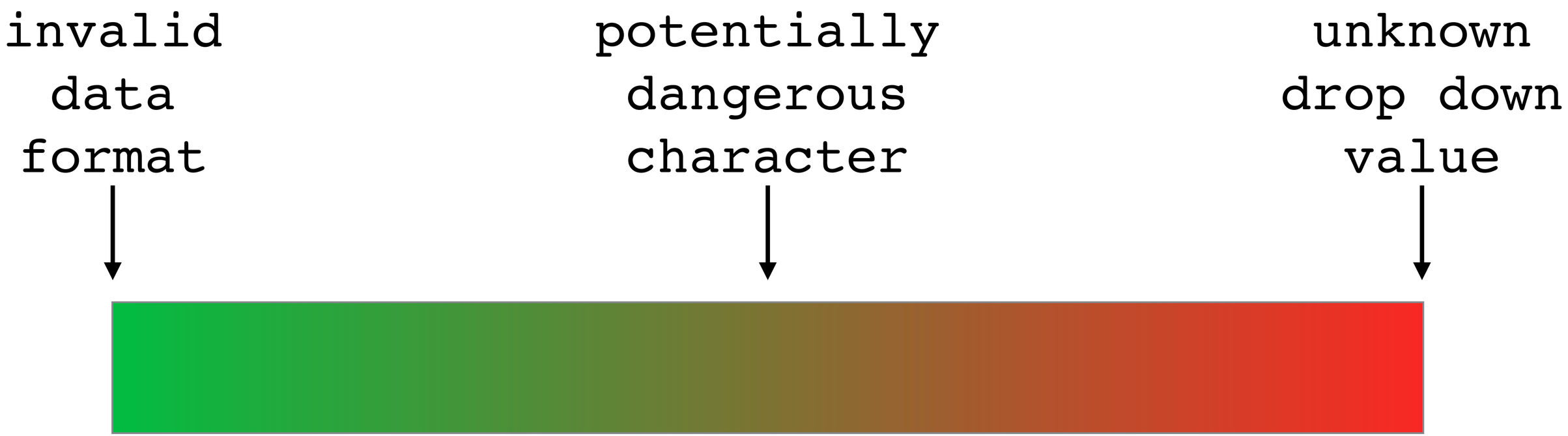
Threshold Examples

2 events over the last 5 minutes

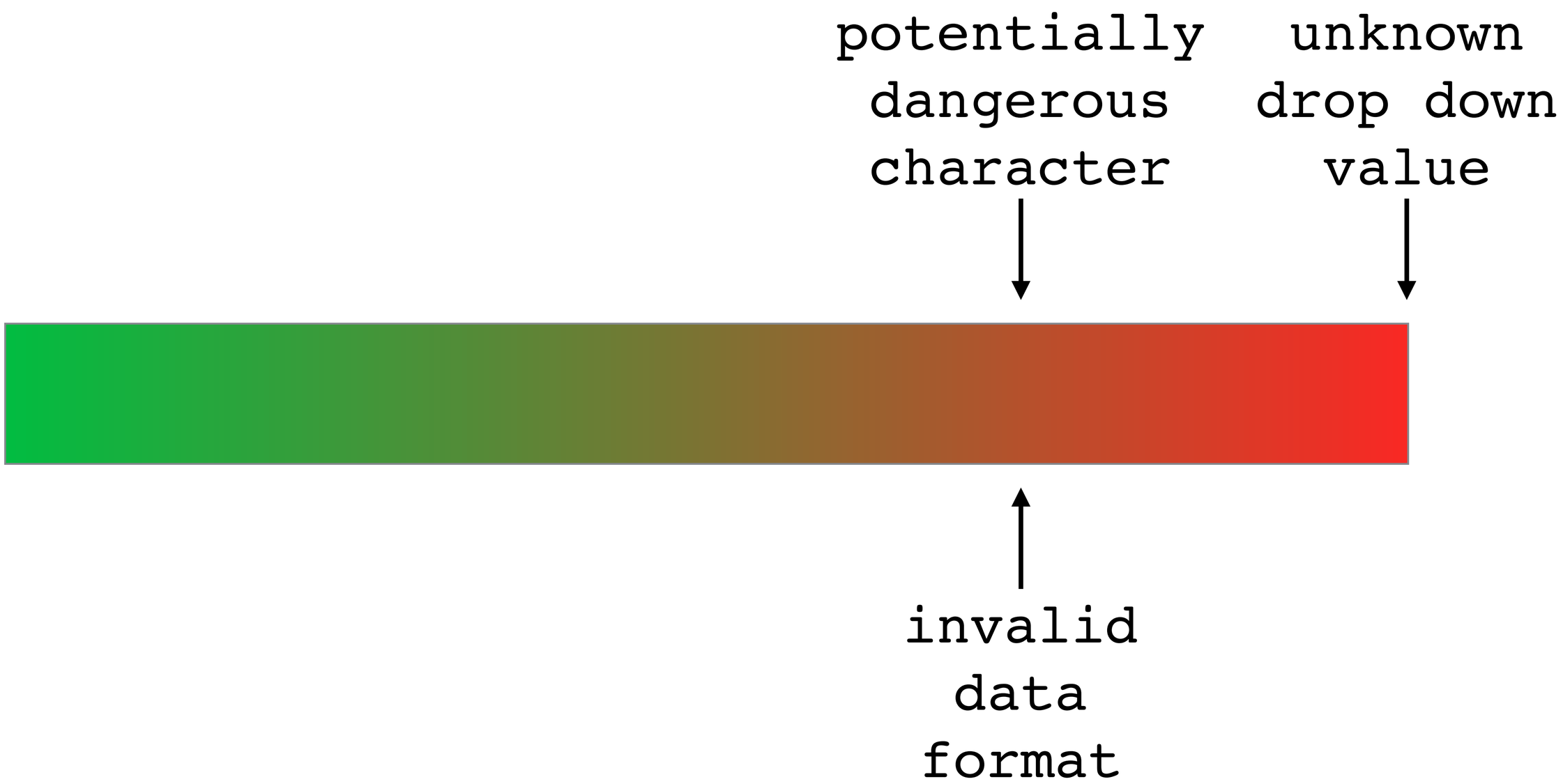
10 events over the last 24 hours

200% increase over one hour

Without Input Validation



With Input Validation



Device and System Tends

More difficult to configure

Require a large device base/ frequent usage

Consider special situations (e.g. weekdays vs. weekends)

Responses

Defend the application and its data

Respond to an attack in any way the application supports

Respond automatically in realtime

No Retaliation

Possible Attack Reactions

No response	Account monitored Admin notification Application monitored	Timing change Request terminated Account logout Account lockout
Function disabled Application disabled	Honeypot redirect	...

React differently on suspicious events and identified attacks

Responses and Thresholds

Threshold	Response
2. event	Log request
4. event	Logout account
6. event	Lockout account

```
<detection-point>
  <category>Input Validation</category>
  <id>IE1</id>
  <threshold>
   <count>2</count>
   <interval unit="minutes">5</interval>
  </threshold>
  <responses>
   <response>
    <action>log</action>
   </response>
   <response>
    <action>logout</action>
   </response>
  </responses>
 </detection-point>
```

```
<detection-point>
  <category>Input Validation</category>
  <id>IE1</id>
  <threshold>
   <count>2</count>
   <interval unit="minutes">5</interval>
  </threshold>
  <responses>
   <response>
    <action>log</action>
   </response>
   <response>
    <action>logout</action>
   </response>
  </responses>
 </detection-point>
```

```
<detection-point>
  <category>Input Validation</category>
  <id>IE1</id>
  <threshold>
   <count>2</count>
   <interval unit="minutes">5</interval>
  </threshold>
  <responses>
   <response>
    <action>log</action>
   </response>
   <response>
    <action>logout</action>
   </response>
  </responses>
 </detection-point>
```

```
<detection-point>
  <category>Input Validation</category>
  <id>IE1</id>
  <threshold>
   <count>2</count>
   <interval unit="minutes">5</interval>
  </threshold>
  <responses>
   <response>
    <action>log</action>
   </response>
   <response>
    <action>logout</action>
   </response>
  </responses>
 </detection-point>
```

```
<detection-point>
  <category>Input Validation</category>
  <id>IE1</id>
  <threshold>
   <count>2</count>
   <interval unit="minutes">5</interval>
  </threshold>
  <responses>
   <response>
    <action>log</action>
   </response>
   <response>
    <action>logout</action>
   </response>
  </responses>
 </detection-point>
```

Implementation

Execution Modes

	REST	SOAP	Thrift	RabbitMQ	Kafka	Embedded
Application language	any					Java
# of applications	multiple					1
Cluster support	yes					no

My Profile arthur@dent.com (Rookie)

Edit Userdata | Change Password

My Encounters

JavaOne 2014 (09/30/2014) 5
San Francisco (USA)

JavaOne 2012 (10/01/2012) 0
San Francisco (USA)

Add Encounter

My Confirmations

JavaOne 2008 (10/10/2012)
San Francisco (USA)

JavaOne 2005 (10/10/2005)
San Francisco (USA)

Add Confirmation

Demo

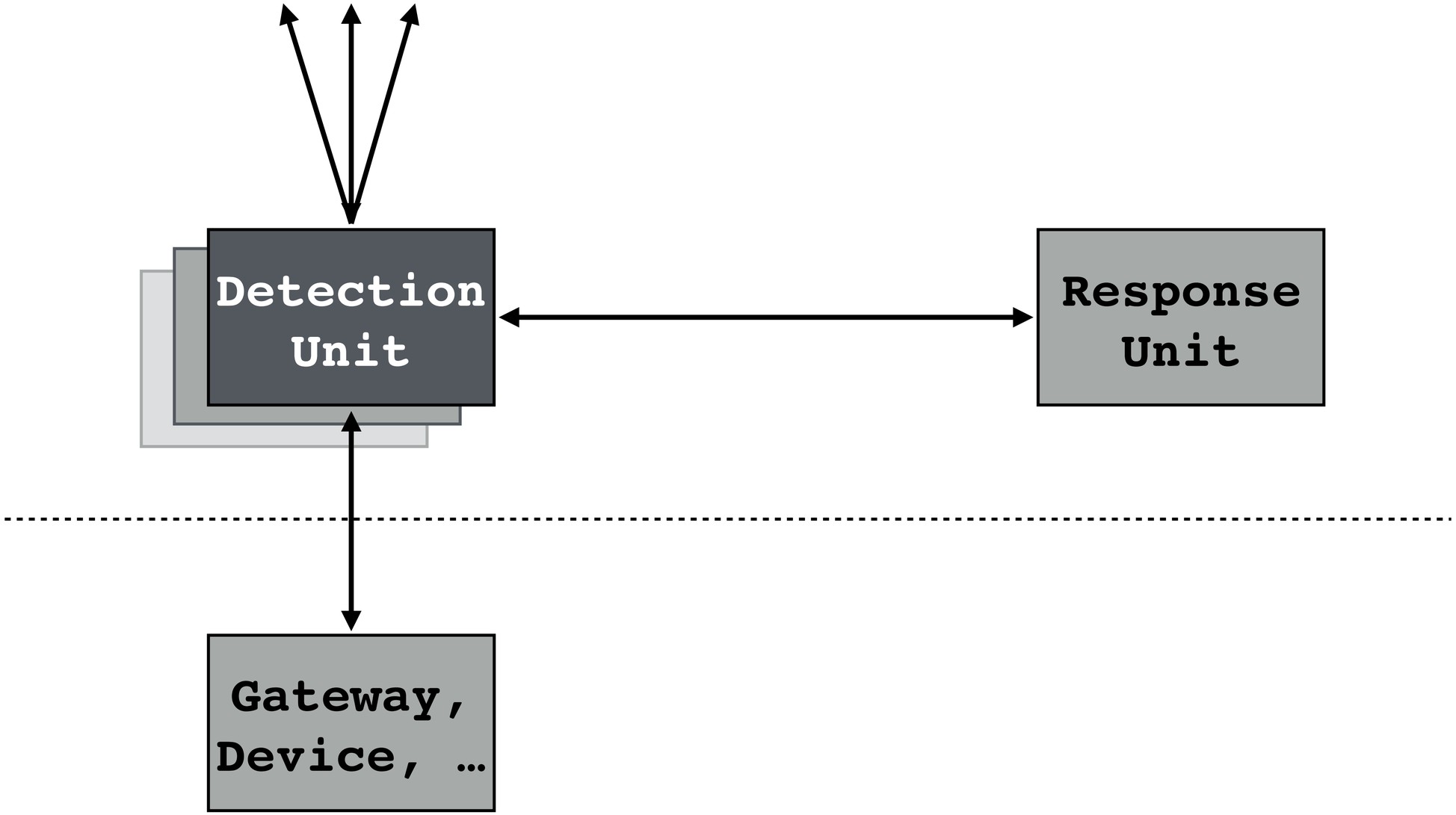
Treat anonymous and authenticated devices differently

```
private void fireEvent() {
 if (user.isAnonymous()) {
  DetectionPoint dp = new DetectionPoint(
   INPUT_VALIDATION, "IE1-001");
  ids.addEvent(new Event(user, dp, detectionSystem));
 } else {
  DetectionPoint dp = new DetectionPoint(
   INPUT_VALIDATION, "IE1-002");
  ids.addEvent(new Event(user, dp, detectionSystem));
 }
}
```

```
private void fireEvent() {
 if (user.isAnonymous()) {
  DetectionPoint dp = new DetectionPoint(
   INPUT_VALIDATION, "IE1-001");
  ids.addEvent(new Event(user, dp, detectionSystem));
 } else {
  DetectionPoint dp = new DetectionPoint(
   INPUT_VALIDATION, "IE1-002");
  ids.addEvent(new Event(user, dp, detectionSystem));
 }
}
```

```
private void fireEvent() {
 if (user.isAnonymous()) {
  DetectionPoint dp = new DetectionPoint(
   INPUT_VALIDATION, "IE1-001");
  ids.addEvent(new Event(user, dp, detectionSystem));
 } else {
  DetectionPoint dp = new DetectionPoint(
   INPUT_VALIDATION, "IE1-002");
  ids.addEvent(new Event(user, dp, detectionSystem));
 }
}
```

Separate Detection and Response Unit(s)



Wrap up

Another layer of defense for your backend

Respond to ongoing attacks in realtime

Reduce impact of successful attacks

Marienstr. 17
70178 Stuttgart

dominik.schadow@bridging-it.de
www.bridging-it.de

Blog blog.dominikschadow.de
Twitter @dschadow

Demo Project
github.com/dschadow/ApplicationIntrusionDetection

OWASP AppSensor
appsensor.org

Detection Points
www.owasp.org/index.php/AppSensor_DetectionPoints

Pictures
www.dreamstime.com