



# Injection-Angriffe: Szenarien, Analyseansätze, Gegenmaßnahmen und Erfahrungen aus der Praxis

Dr. Alexander von Rhein

# Alexander von Rhein

## Research

- Software Verification
- Software-Product-Line Analysis
- Taint-Analysis



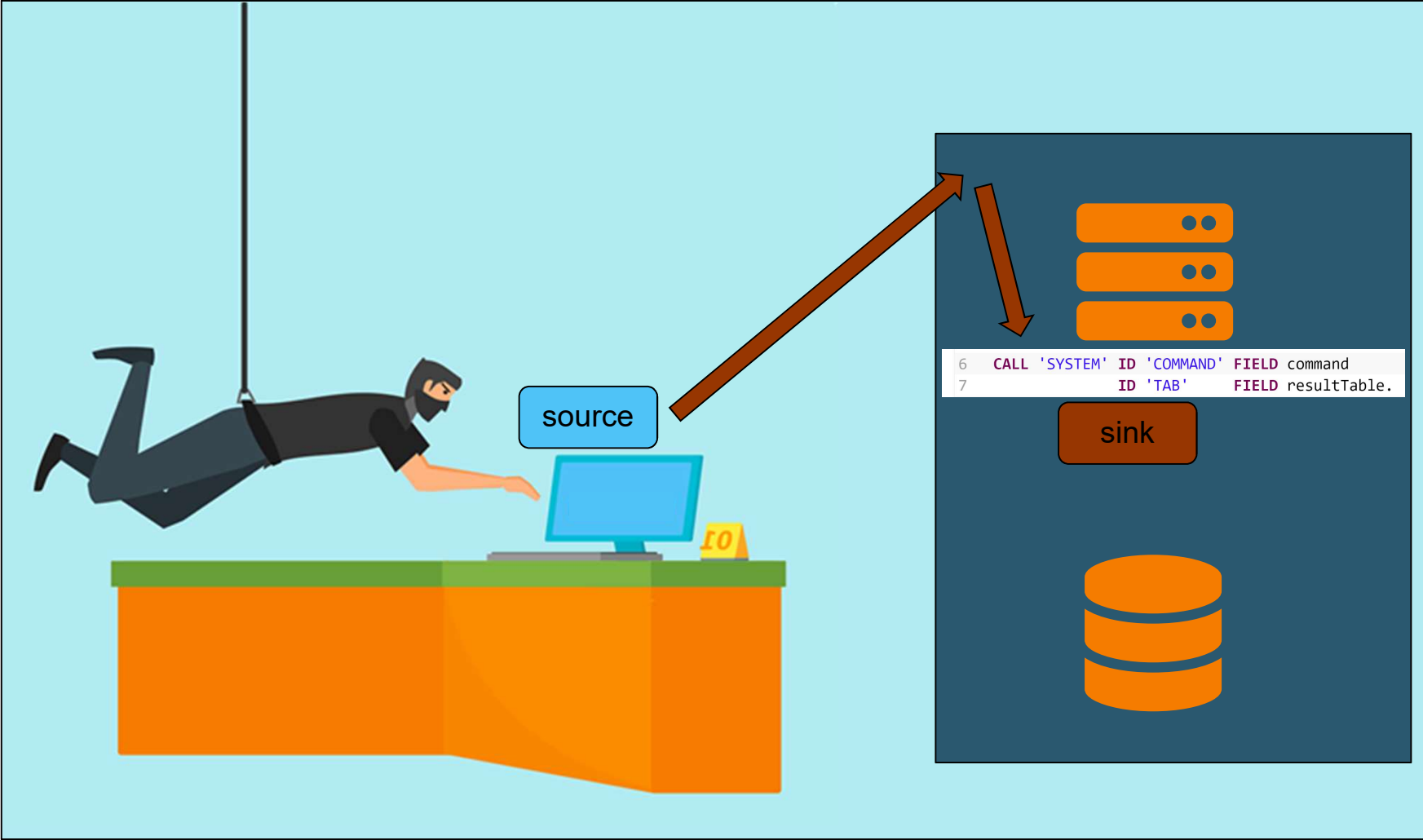
## Consulting

- Software Development
- Quality-Assessment & Quality-Controlling



Free for Research &  
Open Source Projects

# Security Threat: System Command Injection





## Most Common Security Threats in SAP Systems

- Code execution
- Cross-client access
- Directory traversal
- Database modification
- Authentication flaws
- Open SQL injection

Injection/leak attacks

## Security Situation in SAP systems

- 83% Forbes 500 companies use SAP (mainly ERP systems)
- Customization
  - SAP systems are extended with custom code written in ABAP
  - In-house, closed-world development



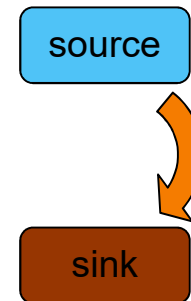
“ *If our company's SAP system is breached it will cost us \$22 million per minute.* ”  
*CISO of a Fortune 500 Company*

[Business Risk Illustration, Onapsis]

# Analysis View – Closed World System

- Source (Report Parameter)
- Sinks
  - System-Commands
    - CALL 'SYSTEM' ID 'COMMAND' FIELD i\_cmd
  - Directory Traversal
    - OPEN DATASET
  - ABAP Program Generation
    - INSERT REPORT
    - GENERATE SUBROUTINE POOL
  - Loop iteration limits
    - DO **input** TIMES. ... ENDDO.
  - ... 21 pattern in total

```
9  PARAMETERS:  
10 p_file TYPE string,  
11 p_table TYPE tabname,  
12 p_code TYPE string.
```



```
12 CALL 'SYSTEM' ID 'COMMAND' FIELD i_cmd  
13 ID 'TAB' FIELD response.
```

```
18 OPEN DATASET i_file FOR OUTPUT  
19 IN TEXT MODE ENCODING DEFAULT.
```

```
40 INSERT REPORT 'ZSECTEST_GENERATED_1' FROM code_lines.
```

```
18 DO p_param TIMES.  
19 ENDDO.
```

# ABAP in 1 minute

Object, can be invoked by user

Parameter of the report

```
1 REPORT zexample.
2 PARAMETERS: p_path TYPE String DEFAULT 'some_path'.
3 CLASS zlcl_example DEFINITION.
4   PUBLIC SECTION.
5     CLASS-METHODS main.
6     CLASS-METHODS compose
7       IMPORTING path TYPE String
8       RETURNING VALUE(returnValue) TYPE String.
9   ENDClass.
10
11 CLASS zlcl_example IMPLEMENTATION.
12   METHOD main.
13     DATA: command TYPE string.
14     command = compose( p_path ).
15     CALL 'SYSTEM' ID 'ls' FIELD command.
16   ENDMETHOD.
17   METHOD compose.
18     returnValue = 'cat ' + path.
19   ENDMETHOD.
20   ENDClass.
21
22 START-OF-SELECTION.
23   zlcl_example=>main( ) .
```

Class declaration

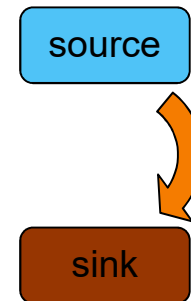
Class implementation

„Main method“ of the report

# Analysis View – Closed World System

- Source (Report Parameter)
- Sinks
  - System-Commands
    - CALL 'SYSTEM' ID 'COMMAND' FIELD i\_cmd
  - Directory Traversal
    - OPEN DATASET
  - ABAP Program Generation
    - INSERT REPORT
    - GENERATE SUBROUTINE POOL
  - Loop iteration limits
    - DO **input** TIMES. ... ENDDO.
  - ... 21 pattern in total

```
9  PARAMETERS:  
10 p_file TYPE string,  
11 p_table TYPE tabname,  
12 p_code TYPE string.
```



```
12 CALL 'SYSTEM' ID 'COMMAND' FIELD i_cmd  
13 ID 'TAB' FIELD response.
```

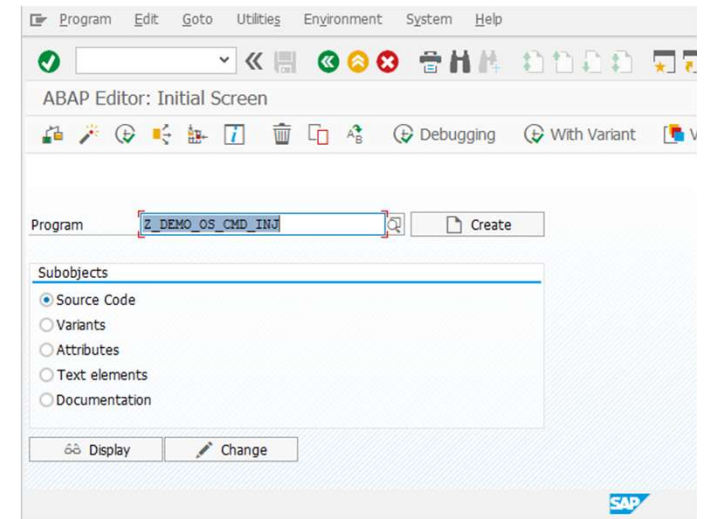
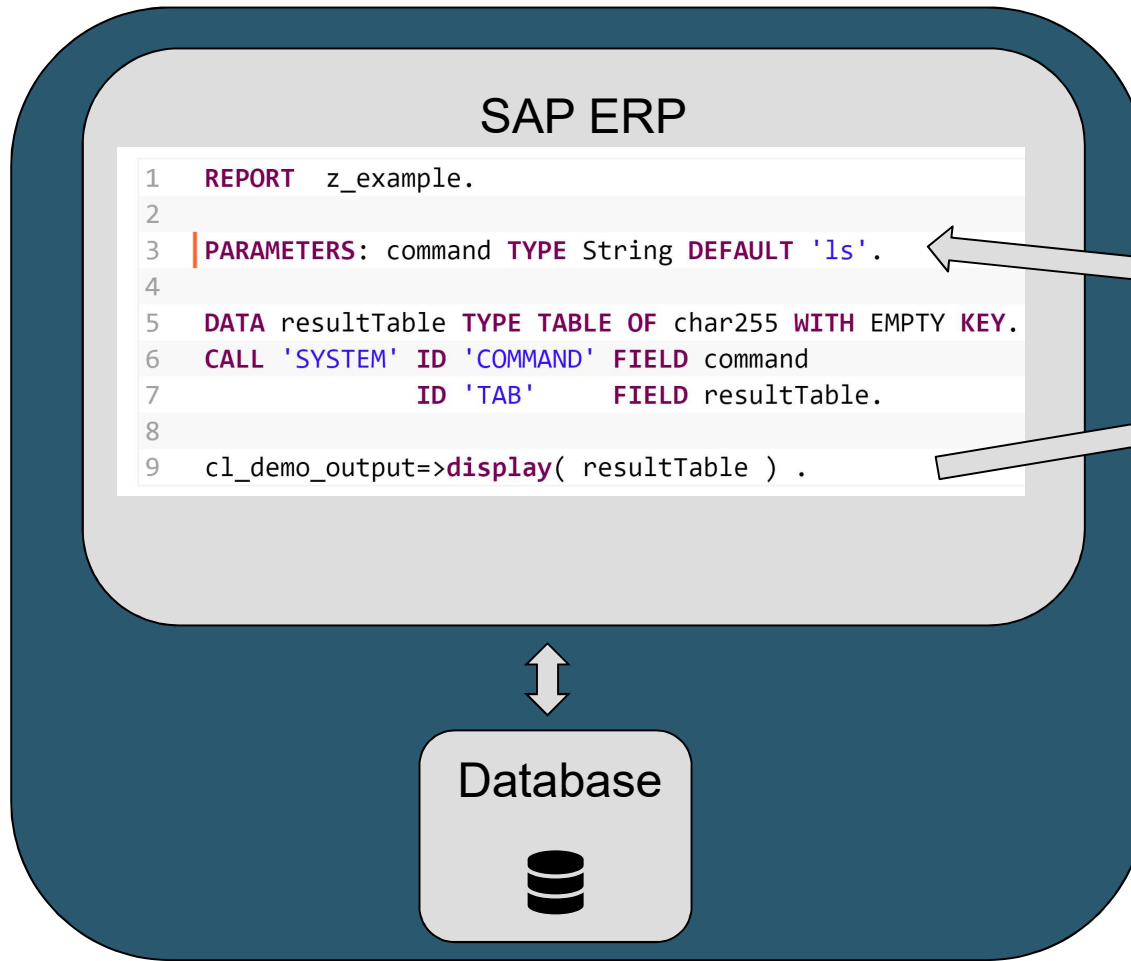
```
18 OPEN DATASET i_file FOR OUTPUT  
19 IN TEXT MODE ENCODING DEFAULT.
```

```
40 INSERT REPORT 'ZSECTEST_GENERATED_1' FROM code_lines.
```

```
18 DO p_param TIMES.  
19 ENDDO.
```

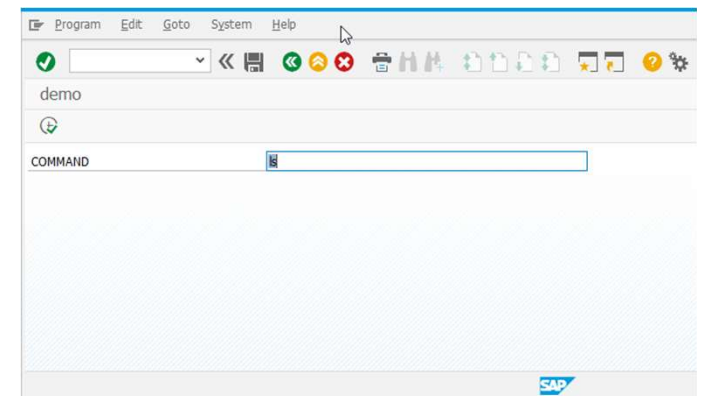


# Simple Security Threat Scenario



**Data Injection**

**Data Leak**



# Trivial Checks for Deprecated Sinks

- Local analysis (typically method level)
- Fast
- Here: Based on discouraged statements

## Usage of CALL 'SYSTEM'

Security / Security

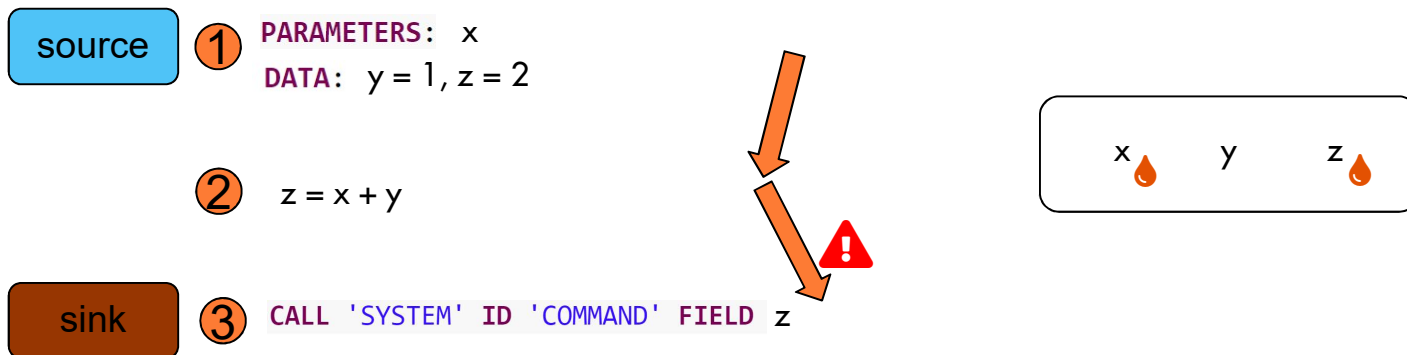
The internal statement `CALL cfunc` executes a system function `cfunc`. Some system functions, e.g. `SYSTEM` or `ThWpInfo` allow to execute any command of the operating system and should be avoided.

The `SXPG framework` should be used instead.

```
25  METHOD main.  
26  DATA: secret TYPE string.  
27  
28  secret = foo( filename ).  
29  secret = secret + 'postfix'.  
30  
31  CALL 'SYSTEM' ID 'ls' FIELD secret.  
32  ENDMETHOD. "main
```

# Taint-Propagation Analysis

- Detailed taint-propagation analysis
  - Requires much more time and memory
  - Data-flow analysis
  - Tracks user input **source** to unsecure statements **sink**



# Global inter-procedural taint-propagation analysis

- Complex data flow source
  - Crossing method boundaries
  - Multiple files
- Large, active code bases
  - Incremental analysis

```
1 REPORT zexample.  
① | PARAMETERS:      p_path TYPE String DEFAULT 'some_path'.  
3 CLASS zlc1_example DEFINITION.  
4   PUBLIC SECTION.  
5     CLASS-METHODS main.  
6     CLASS-METHODS compose  
7       IMPORTING path TYPE String  
8       RETURNING VALUE(returnValue) TYPE String.  
9   ENDClass.  
10  
11 CLASS zlc1_example IMPLEMENTATION.  
12   METHOD main.  
13     DATA: command TYPE string.  
14   ③   command = compose( p_path ).  
15   ⑤   CALL 'SYSTEM' ID 'ls' FIELD command. sink  
16   ENDMETHOD.  
17   METHOD compose.  
18   ④   returnValue = 'cat ' + path.  
19   ENDMETHOD.  
20   ENDClass.  
21  
22 START-OF-SELECTION.  
②   zlc1_example=>main( ) .
```

338-	618-	501-	383-	889-	742-	821-	814-	888-	1078-	1282-	1343-	1323-	1407-	1490-	1574-	1654-	1735-	1820-	1901-	1984-	2064-	2154-	2244-	2334-	2424-	2514-	2604-	2694-	2784-	2874-	2964-	3054-	3144-	3234-	3324-	3414-	3504-	3594-	3684-	3774-	3864-	3954-	4044-	4134-	4224-	4314-	4404-	4494-	4584-	4674-	4764-	4854-	4944-	5034-	5124-	5214-	5304-	5394-	5484-	5574-	5664-	5754-	5844-	5934-	6024-	6114-	6204-	6294-	6384-	6474-	6564-	6654-	6744-	6834-	6924-	7014-	7104-	7194-	7284-	7374-	7464-	7554-	7644-	7734-	7824-	7914-	8004-	8094-	8184-	8274-	8364-	8454-	8544-	8634-	8724-	8814-	8904-	8994-	9084-	9174-	9264-	9354-	9444-	9534-	9624-	9714-	9804-	9894-	9984-	10074-	10164-	10254-	10344-	10434-	10524-	10614-	10704-	10794-	10884-	10974-	11064-	11154-	11244-	11334-	11424-	11514-	11604-	11694-	11784-	11874-	11964-	12054-	12144-	12234-	12324-	12414-	12504-	12594-	12684-	12774-	12864-	12954-	13044-	13134-	13224-	13314-	13404-	13494-	13584-	13674-	13764-	13854-	13944-	14034-	14124-	14214-	14304-	14394-	14484-	14574-	14664-	14754-	14844-	14934-	15024-	15114-	15204-	15294-	15384-	15474-	15564-	15654-	15744-	15834-	15924-	16014-	16104-	16194-	16284-	16374-	16464-	16554-	16644-	16734-	16824-	16914-	17004-	17094-	17184-	17274-	17364-	17454-	17544-	17634-	17724-	17814-	17904-	17994-	18084-	18174-	18264-	18354-	18444-	18534-	18624-	18714-	18804-	18894-	18984-	19074-	19164-	19254-	19344-	19434-	19524-	19614-	19704-	19794-	19884-	19974-	20064-	20154-	20244-	20334-	20424-	20514-	20604-	20694-	20784-	20874-	20964-	21054-	21144-	21234-	21324-	21414-	21504-	21594-	21684-	21774-	21864-	21954-	22044-	22134-	22224-	22314-	22404-	22494-	22584-	22674-	22764-	22854-	22944-	23034-	23124-	23214-	23304-	23394-	23484-	23574-	23664-	23754-	23844-	23934-	24024-	24114-	24204-	24294-	24384-	24474-	24564-	24654-	24744-	24834-	24924-	25014-	25104-	25194-	25284-	25374-	25464-	25554-	25644-	25734-	25824-	25914-	26004-	26094-	26184-	26274-	26364-	26454-	26544-	26634-	26724-	26814-	26904-	26994-	27084-	27174-	27264-	27354-	27444-	27534-	27624-	27714-	27804-	27894-	27984-	28074-	28164-	28254-	28344-	28434-	28524-	28614-	28704-	28794-	28884-	28974-	29064-	29154-	29244-	29334-	29424-	29514-	29604-	29694-	29784-	29874-	29964-	30054-	30144-	30234-	30324-	30414-	30504-	30594-	30684-	30774-	30864-	30954-	31044-	31134-	31224-	31314-	31404-	31494-	31584-	31674-	31764-	31854-	31944-	32034-	32124-	32214-	32304-	32394-	32484-	32574-	32664-	32754-	32844-	32934-	33024-	33114-	33204-	33294-	33384-	33474-	33564-	33654-	33744-	33834-	33924-	34014-	34104-	34194-	34284-	34374-	34464-	34554-	34644-	34734-	34824-	34914-	35004-	35094-	35184-	35274-	35364-	35454-	35544-	35634-	35724-	35814-	35904-	36000-	36094-	36184-	36274-	36364-	36454-	36544-	36634-	36724-	36814-	36904-	37000-	37094-	37184-	37274-	37364-	37454-	37544-	37634-	37724-	37814-	37904-	38000-	38094-	38184-	38274-	38364-	38454-	38544-	38634-	38724-	38814-	38904-	39000-	39094-	39184-	39274-	39364-	39454-	39544-	39634-	39724-	39814-	39904-	40000-	40094-	40184-	40274-	40364-	40454-	40544-	40634-	40724-	40814-	40904-	41000-	41094-	41184-	41274-	41364-	41454-	41544-	41634-	41724-	41814-	41904-	42000-	42094-	42184-	42274-	42364-	42454-	42544-	42634-	42724-	42814-	42904-	43000-	43094-	43184-	43274-	43364-	43454-	43544-	43634-	43724-	43814-	43904-	44000-	44094-	44184-	44274-	44364-	44454-	44544-	44634-	44724-	44814-	44904-	45000-	45094-	45184-	45274-	45364-	45454-	45544-	45634-	45724-	45814-	45904-	46000-	46094-	46184-	46274-	46364-	46454-	46544-	46634-	46724-	46814-	46904-	47000-	47094-	47184-	47274-	47364-	47454-	47544-	47634-	47724-	47814-	47904-	48000-	48094-	48184-	48274-	48364-	48454-	48544-	48634-	48724-	48814-	48904-	49000-	49094-	49184-	49274-	49364-	49454-	49544-	49634-	49724-	49814-	49904-	50000-	50094-	50184-	50274-	50364-	50454-	50544-	50634-	50724-	50814-	50904-	51000-	51094-	51184-	51274-	51364-	51454-	51544-	51634-	51724-	51814-	51904-	52000-	52094-	52184-	52274-	52364-	52454-	52544-	52634-	52724-	52814-	52904-	53000-	53094-	53184-	53274-	53364-	53454-	53544-	53634-	53724-	53814-	53904-	54000-	54094-	54184-	54274-	54364-	54454-	54544-	54634-	54724-	54814-	54904-	55000-	55094-	55184-	55274-	55364-	55454-	55544-	55634-	55724-	55814-	55904-	56000-	56094-	56184-	56274-	56364-	56454-	56544-	56634-	56724-	56814-	56904-	57000-	57094-	57184-	57274-	57364-	57454-	57544-	57634-	57724-	57814-	57904-	58000-	58094-	58184-	58274-	58364-	58454-	58544-	58634-	58724-	58814-	58904-	59000-	59094-	59184-	59274-	59364-	59454-	59544-	59634-	59724-	59814-	59904-	60000-	60094-	60184-	60274-	60364-	60454-	60544-	60634-	60724-	60814-	60904-	61000-	61094-	61184-	61274-	61364-	61454-	61544-	61634-	61724-	61814-	61904-	62000-	62094-	62184-	62274-	62364-	62454-	62544-	62634-	62724-	62814-	62904-	63000-	63094-	63184-	63274-	63364-	63454-	63544-	63634-	63724-	63814-	63904-	64000-	64094-	64184-	64274-	64364-	64454-	64544-	64634-	64724-	64814-	64904-	65000-	65094-	65184-	65274-	65364-	65454-	65544-	65634-	65724-	65814-	65904-	66000-	66094-	66184-	66274-	66364-	66454-	66544-	66634-	66724-	66814-	66904-	67000-	67094-	67184-	67274-	67364-	67454-	67544-	67634-	67724-	67814-	67904-	68000-	68094-	68184-	68274-	68364-	68454-	68544-	68634-	68724-	68814-	68904-	69000-	69094-	69184-	69274-	69364-	69454-	69544-	69634-	69724-	69814-	69904-	70000-	70094-	70184-	70274-	70364-	70454-	70544-	70634-	70724-	70814-	70904-	71000-	71094-	71184-	71274-	71364-	71454-	71544-	71634-	71724-	71814-	71904-	72000-	72094-	72184-	72274-	72364-	72454-	72544-	72634-	72724-	72814-	72904-	73000-	73094-	73184-	73274-	73364-	73454-	73544-	73634-	73724-	73814-	73904-	74000-	74094-	74184-	74274-	74364-	74454-	74544-	74634-	74724-	74814-	74904-	75000-	75094-	75184-	75274-	75364-	75454-	75544-	75634-	75724-	75814-	75904-	76000-	76094-	76184-	76274-	76364-	76454-	76544-	76634-	76724-	76814-	76904-	77000-	77094-	77184-	77274-	77364-	77454-	77544-	77634-	77724-	77814-	77904-	78000-	78094-	78184-	78274-	78364-	78454-	78544-	78634-	78724-	78814-	78904-	79000-	79094-	79184-	79274-	79364-	79454-	79544-	79634-	79724-	79814-	79904-	80000-	80094-	80184-	80274-	80364-	80454-	80544-	80634-	80724-	80814-	80904-	81000-	81094-	81184-	81274-	81364-	81454-	81544-	81634-	81724-	81814-	81904-	82000-	82094-	82184-	82274-	82364-	82454-	82544-	82634-	82724-	82814-	82904-	83000-	83094-	83184-	83274-	83364-	83454-	83544-	83634-	83724-	83814-	83904-	84000-	84094-	84184-	84274-	84364-	84454-	84544-	84634-	84724-	84814-	84904-	85000-	85094-	85184-	85274-	85364-	85454-	85544-	85634-	85724-	85814-	85904-	86000-	86094-	86184-	86274-	86364-	86454-	86544-	86634-	86724-	86814-	86904-	87000-	87094-	87184-	87274-	87364-	87454-	87544-	87634-	87724-	87814-	87904-	88000-	88094-	88184-	88274-	88364-	88454-	88544-	88634-	88724-	88814-	88904-	89000-	89094-	89184-	89274-	89364-	89454-	89544-	89634-	89724-	89814-	89904-	90000-	90094-	90184-	90274-	90364-	90454-	90544-	90634-	90724-	90814-	90904-	91000-	91094-	91184-	91274-	91364-	91454-	91544-	91634-	91724-	91814-	91904-	92000-	92094-	92184-	92274-	92364-	92454-	92544-	92634-	92724-	92814-	92904-	93000-	93094-	93184-	93274-	93364-	93454-	93544-	93634-	93724-	93814-	93904-	94000-	94094-	94184-	94274-	94364-	94454-	94544-	94634-	94724-	94814-	94904-	95000-	95094-	95184-	95274-	95364-	95454-	95544-	95634-	95724-	95814-	95904-	96000-	96094-	96184-	96274-	96364-	96454-	96544-	96634-	96724-	96814-	96904-	97000-	97094-	97184-	97274-	97364-	97454-	97544-	97634-	97724-	97814-	97904-	98000-	98094-	98184-	98274-	98364-	98454-	98544-	98634-	98724-	98814-	98904-	99000-	99094-	99184-	99274-	99364-	99454-	99544-	99634-	99724-	99814-	99904-	100000-	100094-	100184-	100274-	100364-	100454-	100544-	100634-	100724-	100814-	100904-	101000-	101094-	101184-	101274-	101364-	101454-	101544-	101634-	101724-	101814-	101904-	102000-	102094-	102184-	102274-	102364-	102454-	102544-	102634-	102724-	102814-	102904-	103000-	103094-	103184-	103274-	103364-	103454-	103544-	103634-	103724-	103814-	103904-	104000-	104094-	104184-	104274-	104364-	104454-	104544-	104634-	104724-	104814-	104904-	105000-	105094-	105184-	105274-	105364-	105454-	105544-	105634-	105724-	105814-	105904-	106000-	106094-	106184-	106274-	106364-	106454-	106544-	106634-	106724-	106814-	106904-	107000-	107094-	107184-	107274-	107364-	107454-	107544-	107634-	107724-	107814-	107904-	108000-	108094-	108184-	108274-	108364-	108454-	108544-	108634-	108724-	108814-	108904-	109000-	109094-	109184-	109274-	109364-	109454-	109544-	109634-	109724-	109814-	109904-	110000-	110094-	110184-	110274-	110364-	110454-	110544-	110634-	110724-	110814-	110904-	111000-	111094-	111184-	111274-	111364-	111454-	111544-	111634-	111724-	111814-	111
------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	-----

# Findings im Benchmark

- Trivial Checks
  - 12.943 yellow findings
  - 24.555 red findings
- Taint Analysis
  - 7.251 taint-analysis findings
  - Some methods (2%) had to be ignored (cycles, complexity)

## Missing authority check before CALL TRANSACTION

Code Anomalies / Security

CALL TRANSACTION executes an transaction with the given transaction code. Until SAP\_BASIS release 7.40 no authority check was performed for CALL TRANSACTION, from SAP\_BASIS release 7.40 on, a authority check is only performed, if the addition WITH AUTHORITY CHECK is used.

To solve this issue, the addition WITH AUTHORITY CHECK should be added (if SAP\_BASIS version is 7.40 or higher) or the function module AUTHORITY\_CHECK\_TCODE should be called before.

```
in PROGRAMMER_XXX...
...
536 CALL TRANSACTION USING MESSAGE_MODE
537 MESSAGES INTO MESSAGE.
```



## Execution of OS command: filename reaching secret at Z\_CALLER.abap line 31.

taint analysis / Taint Propagation

A data value is read from user input and passed to a statement that executes OS commands.

For further information see [SAP help page on system command injection](#).

- Source: PROG/Z\_CALLER.abap:3
- Sink: PROG/Z\_CALLER.abap:31

# Performance



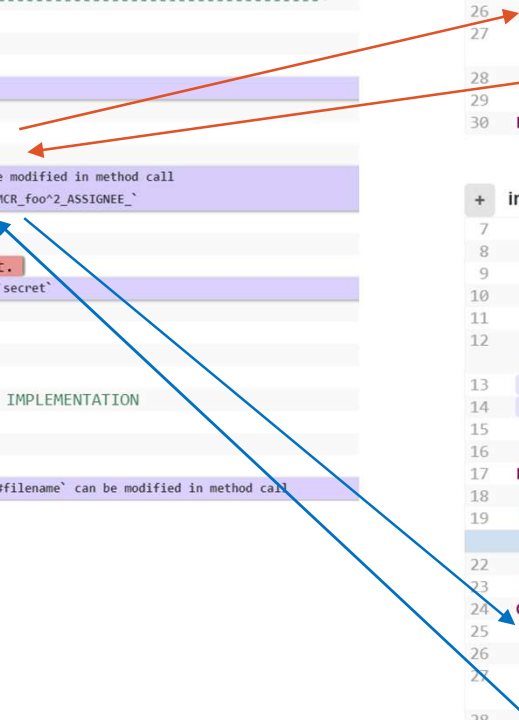
- Performance benchmark
  - 12.600.000 source lines of code from customers
    - Some projects use git, so they have actually more code
  - with 270.000 methods
- 3 hours initial analysis time
- Analysis time for single commit depends on number of „touched“ methods
  - Typically few seconds

in PROG/Z\_CALLER.abap:3

```
+ in PROG/Z_CALLER.abap
1 REPORT ztestsrc_df_class.
2
3 PARAMETERS: filename TYPE String DEFAULT 'default_filename'.
4 [0] This source taints the variable `filename`
5
6 *-----*
7 * CLASS zlc1_main DEFINITION
8 *
9 *
10 *-----*
11 *
12 *-----*
13 *
14 *-----*
15 *
16 *-----*
17 *
18 *-----*
19 *
20 *-----*
21 *
22 *-----*
23 CLASS zlc1_main IMPLEMENTATION.
24
25 METHOD main. ]
26 [2] Field `filename` may contain tainted data
27 DATA: secret TYPE string.
28
29 secret = foo( filename ). ]
30 [3] Possibly tainted parameter at index 0 can be modified in method call
31 [6] Possibly tainted return value in assignee `MCR_foo^2_ASSIGNEE_`
32 secret = secret + 'postfix'.
33
34 CALL 'SYSTEM' ID 'ls' FIELD secret.
35 [7] OS command injection with tainted variable `secret`
36
37 ENDMETHOD. "main
38
39 METHOD foo.
40
41 RETURN VALUE = 'CLEAN'.
42
43 ENDMETHOD. "main
44
45 ENDClass. "lcl_main IMPLEMENTATION
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

```
+ in CLAS/ZCL_UNSAFE_TEST.abap
7 PUBLIC
8 CREATE PUBLIC .
9
10 PUBLIC SECTION.
11
12 [METHODS foo ]
13 [4] Possibly tainted parameter `info`
14 IMPORTING info TYPE String ]
15 RETURNING value(r_result) TYPE string . ]
16 PROTECTED SECTION.
17 PRIVATE SECTION.
18 ENDClass. "ZCL_UNSAFE_TEST DEFINITION
19 *-----*
20 *
21 *-----*
22 *
23 *-----*
24 CLASS zcl_unsafe_test IMPLEMENTATION.
25
26 METHOD foo.
27 r_result = info. ]
28 [5] Variable `r_result` may be tainted.
29 ENDMETHOD. "foo
30 ENDClass. "ZCL_UNSAFE_TEST IMPLEMENTATION
```

```
+ in CLAS/ZCL_UNSAFE_TEST_WITH_SINK.abap
7 PUBLIC
8 CREATE PUBLIC .
9
10 PUBLIC SECTION.
11
12 [METHODS foo ]
13 [8] Possibly tainted parameter `info`
14 IMPORTING info TYPE String ]
15 RETURNING value(r_result) TYPE string . ]
16 PROTECTED SECTION.
17 PRIVATE SECTION.
18 ENDClass. "ZCL_SAFE_TEST DEFINITION
19 *-----*
20 *
21 *-----*
22 *
23 *-----*
24 CLASS zcl_safe_test IMPLEMENTATION.
25
26 METHOD foo.
27 CALL 'SYSTEM' ID 'ls' FIELD info.
28 r_result = info. ]
29 [9] Variable `r_result` may be tainted.
30 ENDMETHOD. "main
31 ENDClass. "ZCL_SAFE_TEST IMPLEMENTATION
```







## Beyond ABAP

- ABAP
  - Closed world
  - Client/Server setting
  - Database and Server-Filesystem are typically trusted
  - Entropy of identifiers (method names, variable names) is high
- Java, C#, ...
  - No closed-world scenario
  - Who defines the taint sources and sinks?
  - More use of high-level programming (inheritance, lambdas, ...)
  - Many similar variable and method names



Code



Models



Version History



Static Analysis

cmocka

**JUnit**

Test Results



Issue Trackers



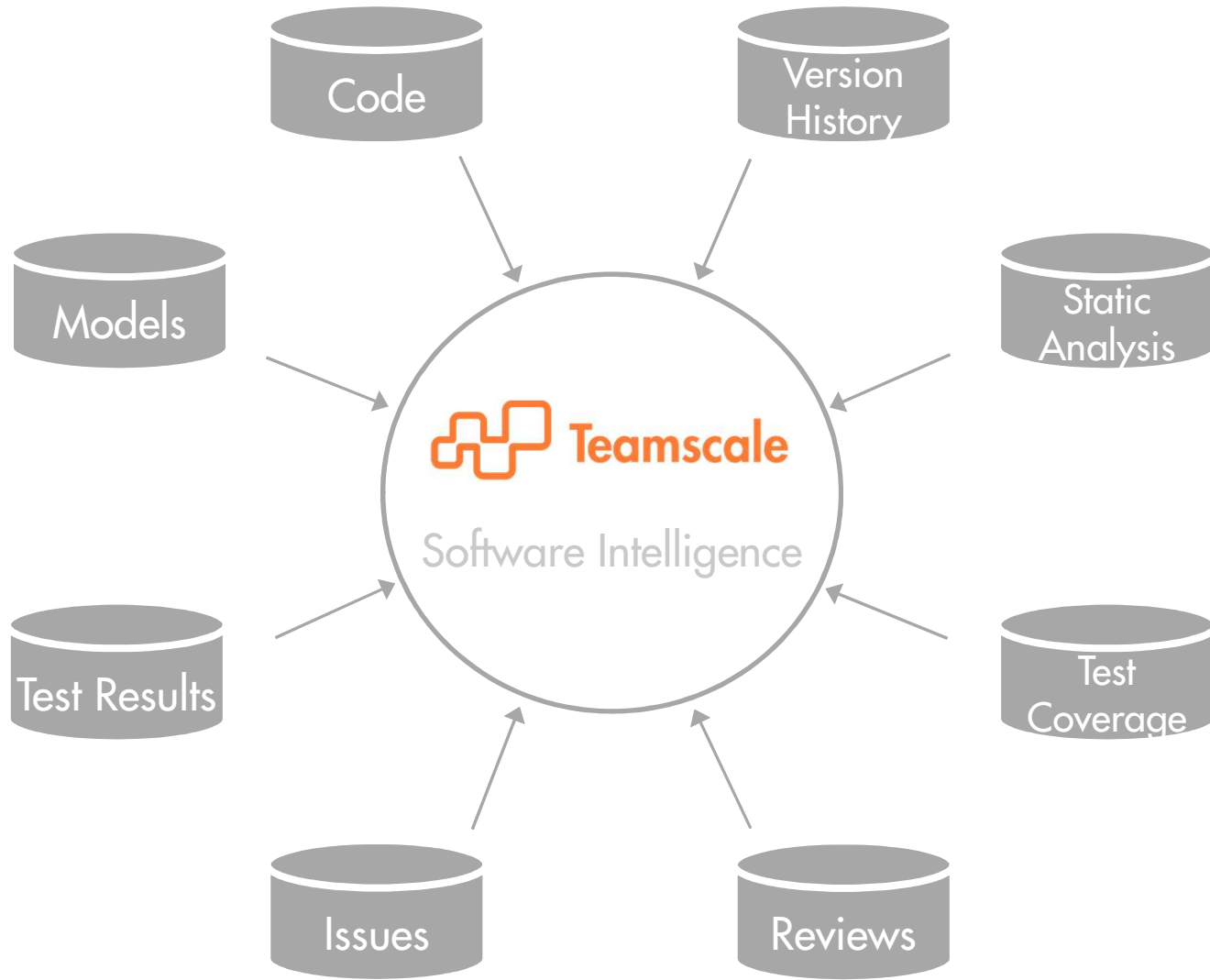
Reviews

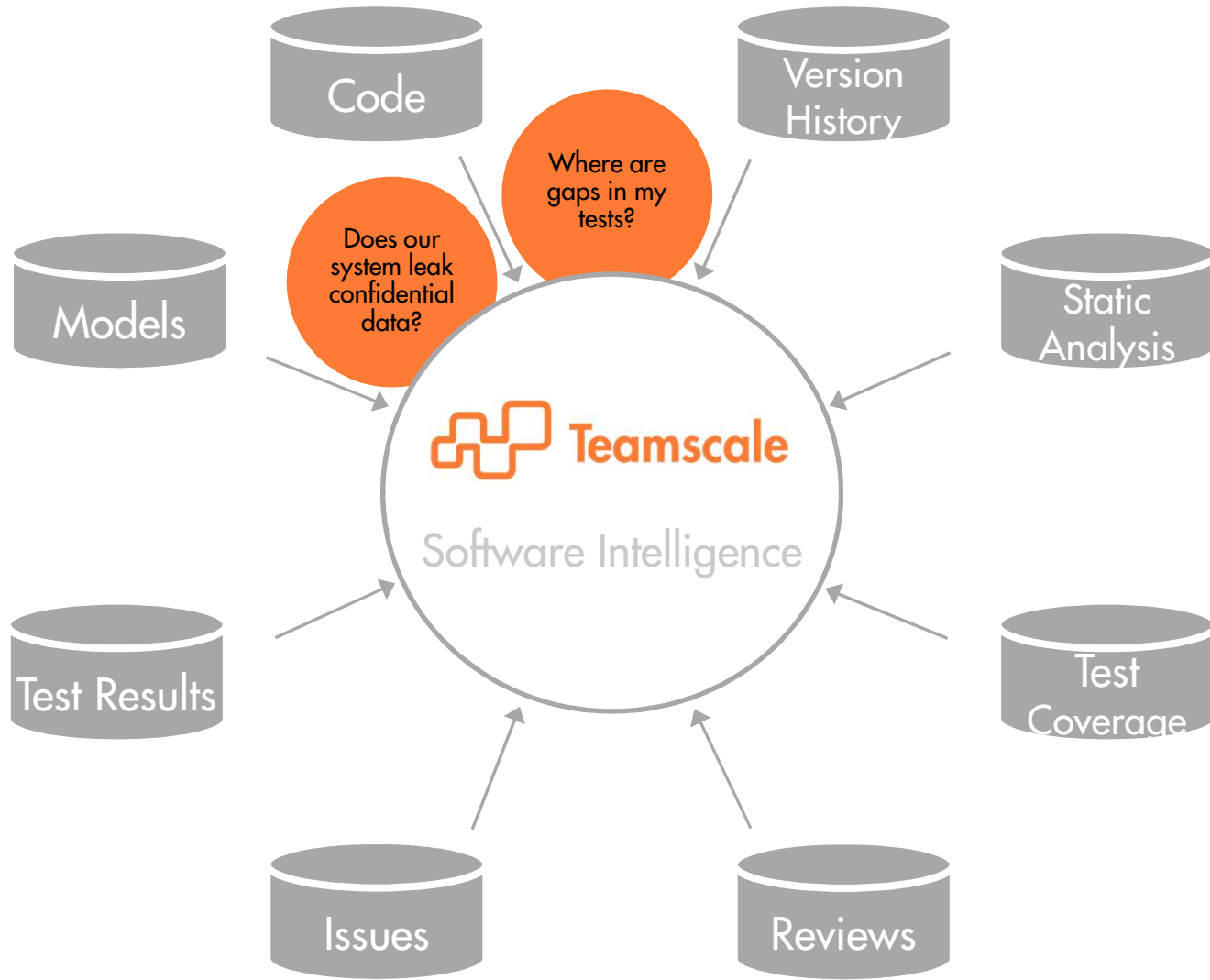
**GCOV**



Test Coverage

... and many more.





GUI.Dialogs

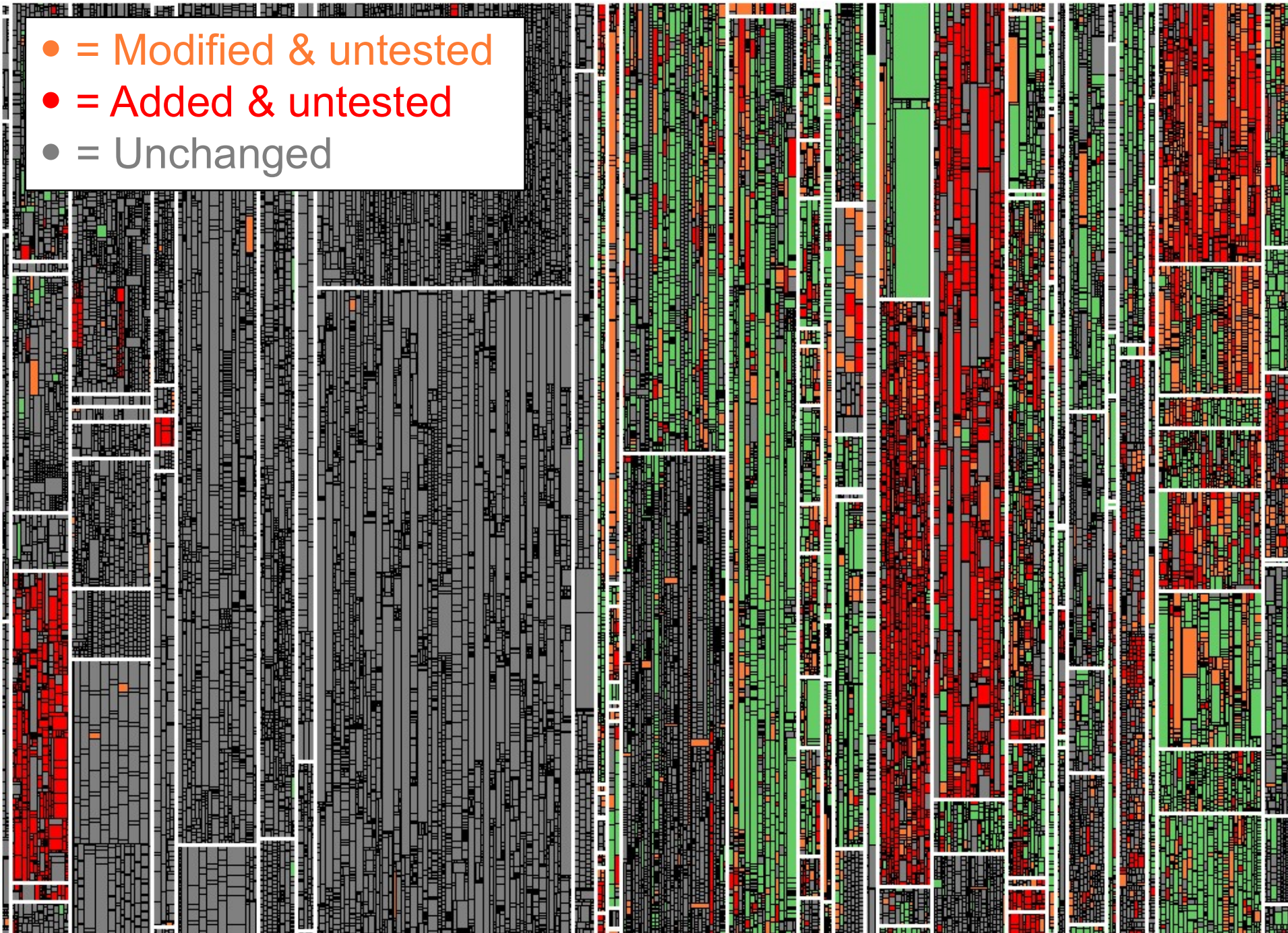
UI.Controls

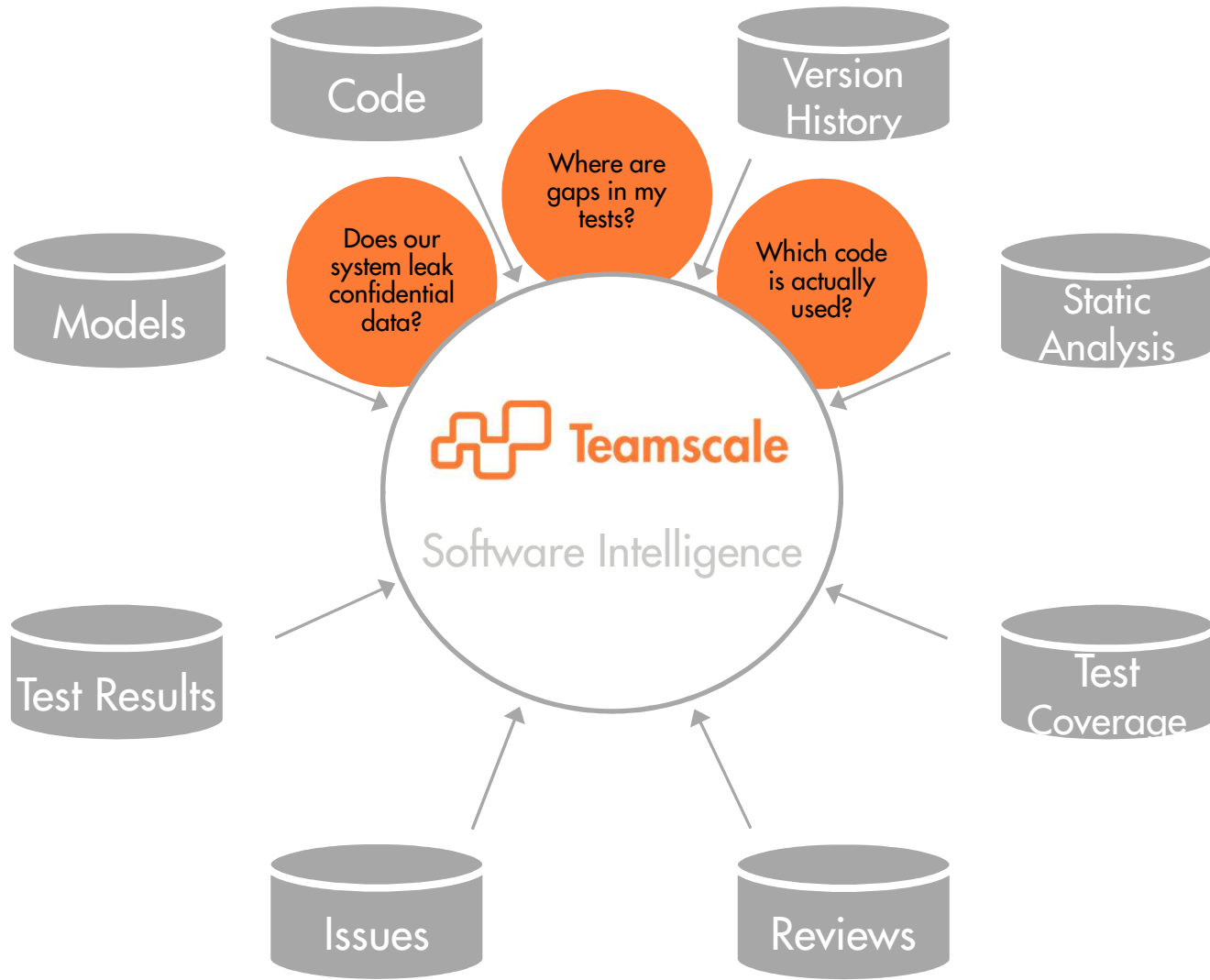
Authentication

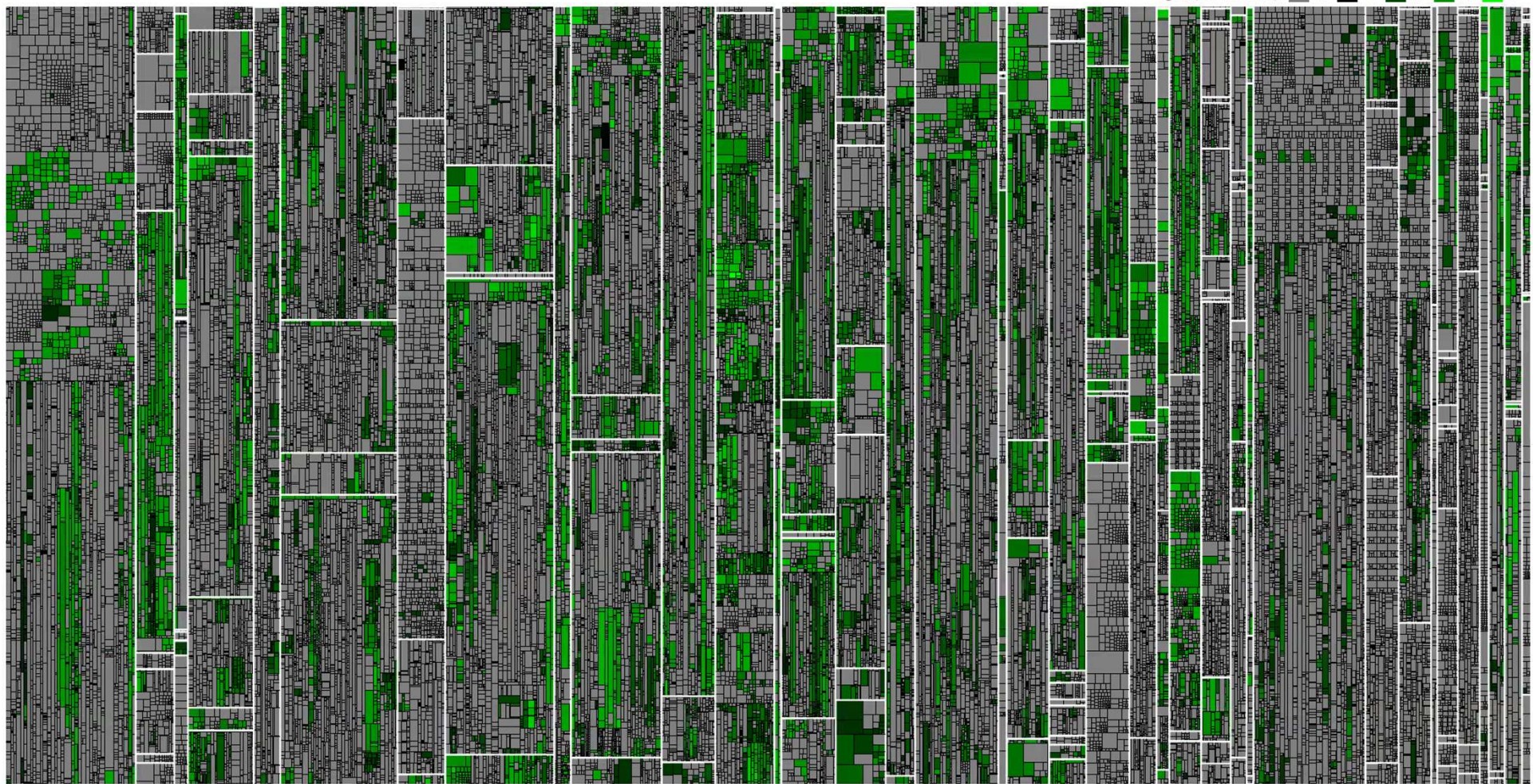
GUI.Base

Data  
Validation

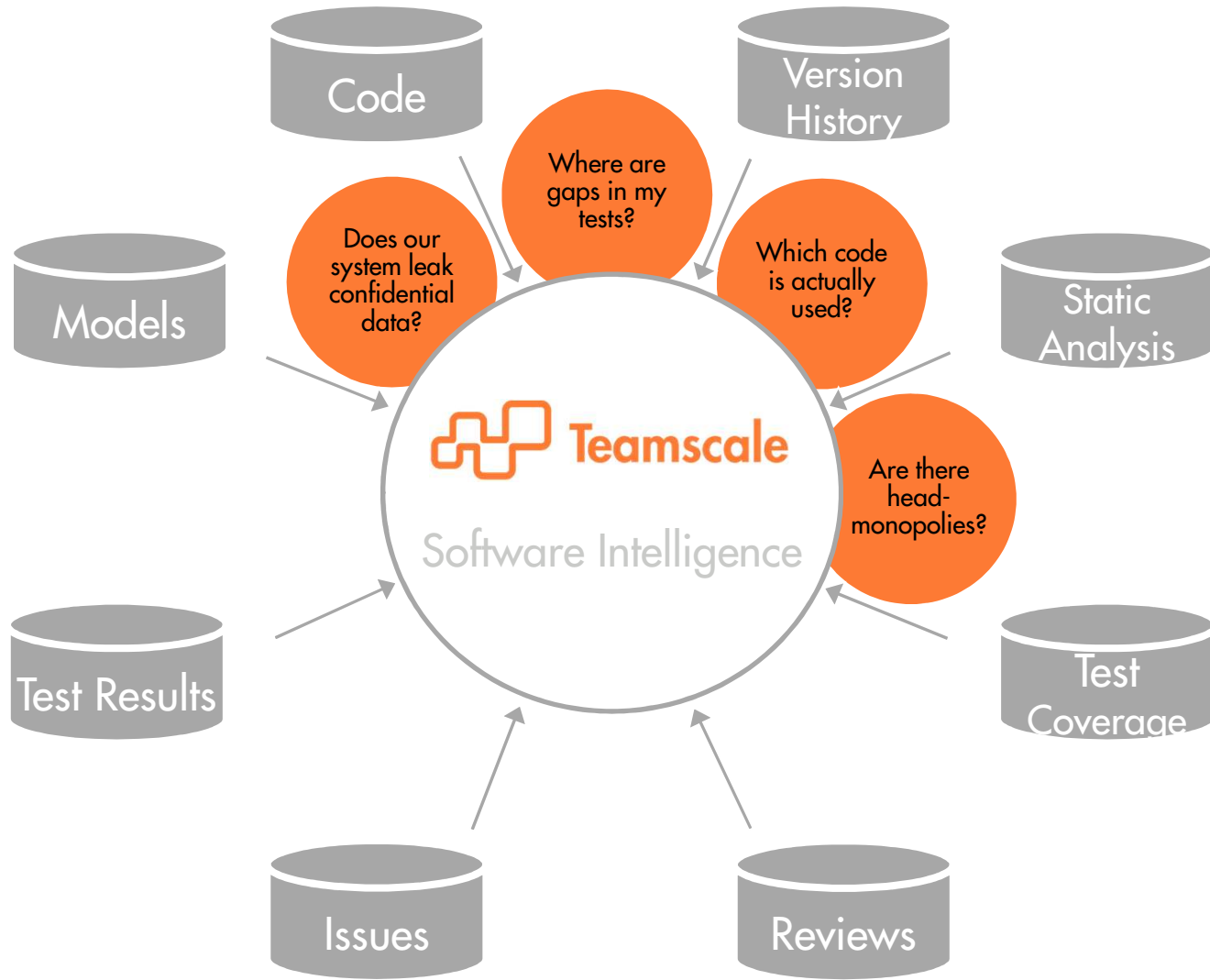
- = Modified & untested
- = Added & untested
- = Unchanged

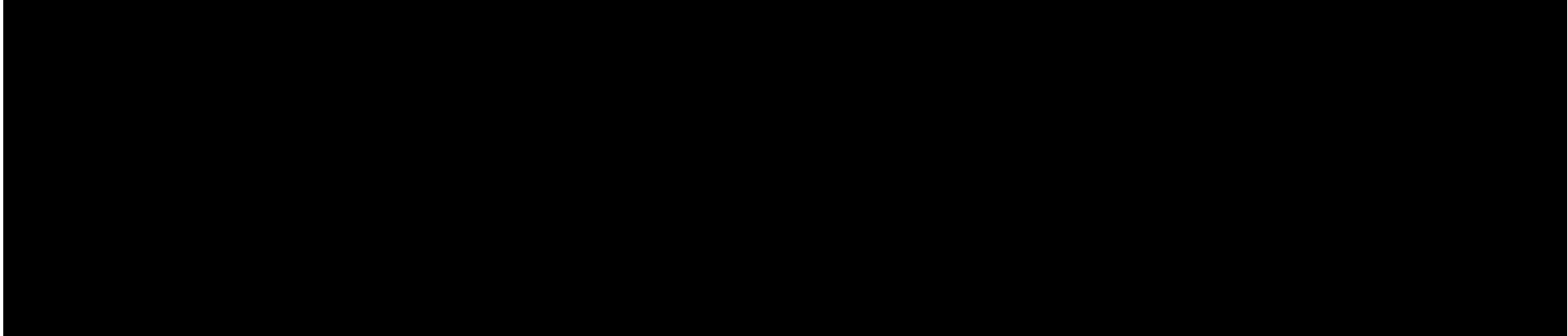
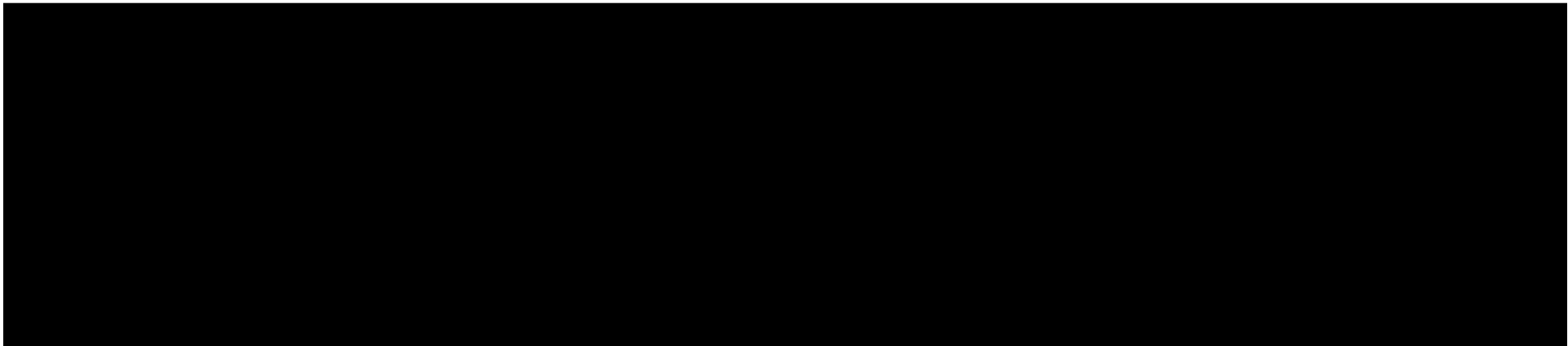


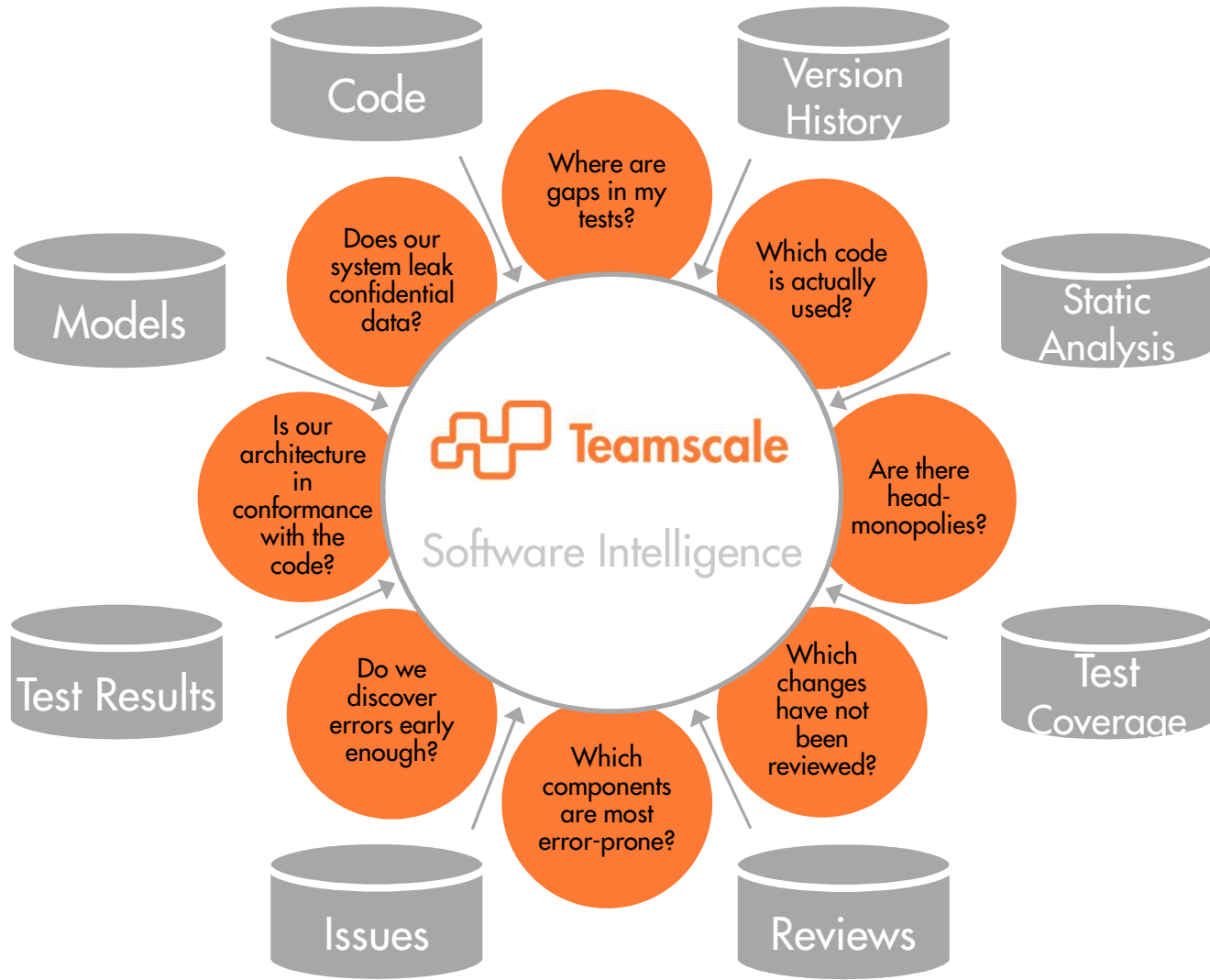












## Incremental global ABAP security analysis with Teamscale

Posted on 11/10/2017 by [Dr. Alexander von Rhein](#)

Our mission at CQSE is to help customers improve the quality of their code. Our tool **Teamscale** checks source code and reports issues such as logical flaws, copy&paste programming and possible performance bottlenecks. However there is one aspect of code quality that we did not address so far: code security. Code is secure if it can not be used by an attacker to perform unintended, dangerous actions on the host system. For example, if an attacker inserts `' ; DROP TABLE Customers;` in an input field, this might cause a system to delete the Customers table—a well known »SQL Injection« attack. In this post, I will explain how new analyses in Teamscale can efficiently detect vulnerabilities for such attacks and report them to developers.



### Security Threats in Source Code



## Conclusion

Static analysis can find many attack scenarios at development time.

Security attacks are often injection/leak attacks.

(Near) real-time feedback is vital for acceptance. Our solution is incremental analysis.

Wanted: Evaluation partners for security analyses (and teamscale in general).

## Kontakt

Dr. Alexander von Rhein · [rhein@cqse.eu](mailto:rhein@cqse.eu) · +49 159 04517754  
@alexvonrhein  
[www.cqse.eu/en/blog](http://www.cqse.eu/en/blog)



CQSE GmbH  
Centa-Hafenbrädl-Straße 59  
81249 München