SAST снаружи и изнутри

Сергей Васильев

Expert

Павел Еремеев

CTO в PVS-Studio

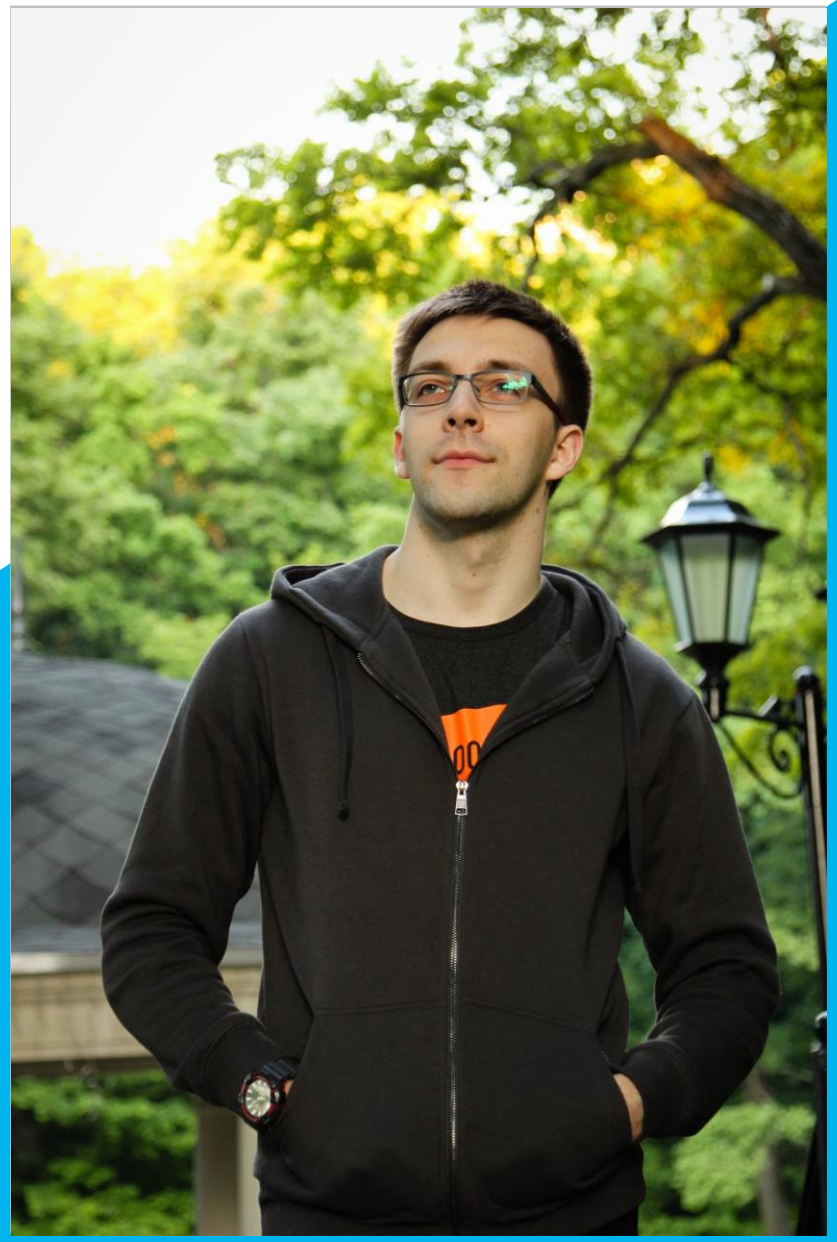
Помогает команде делать качественный продукт

Speaker

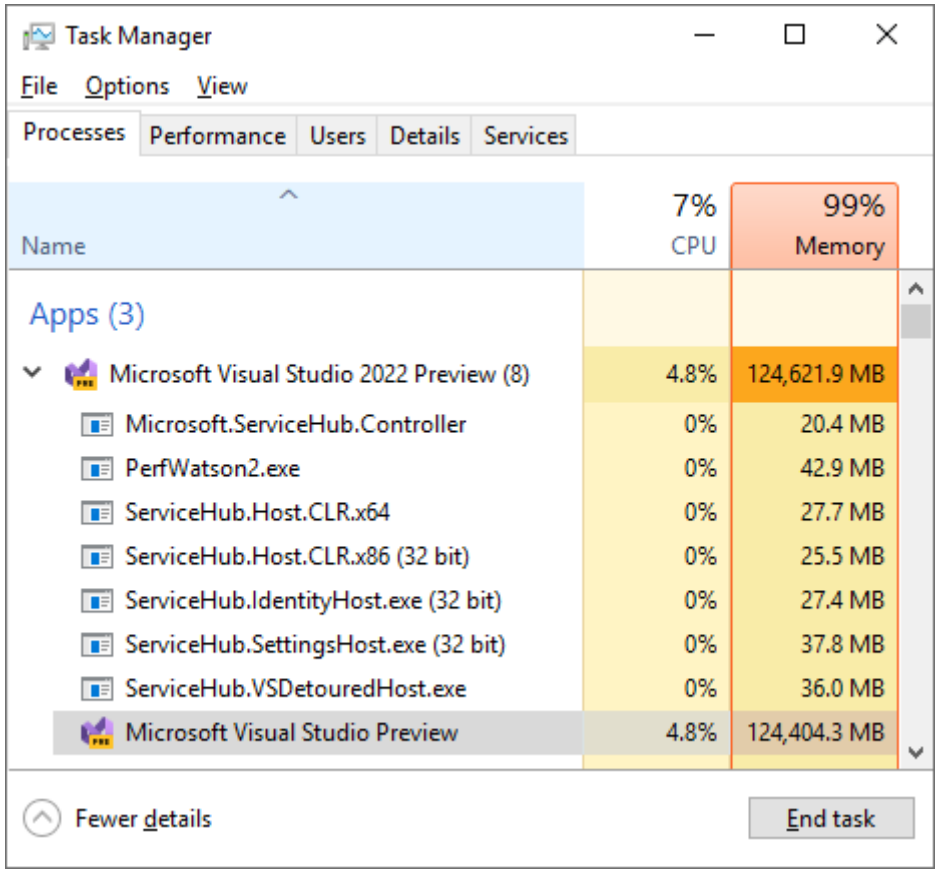
Сергей Васильев

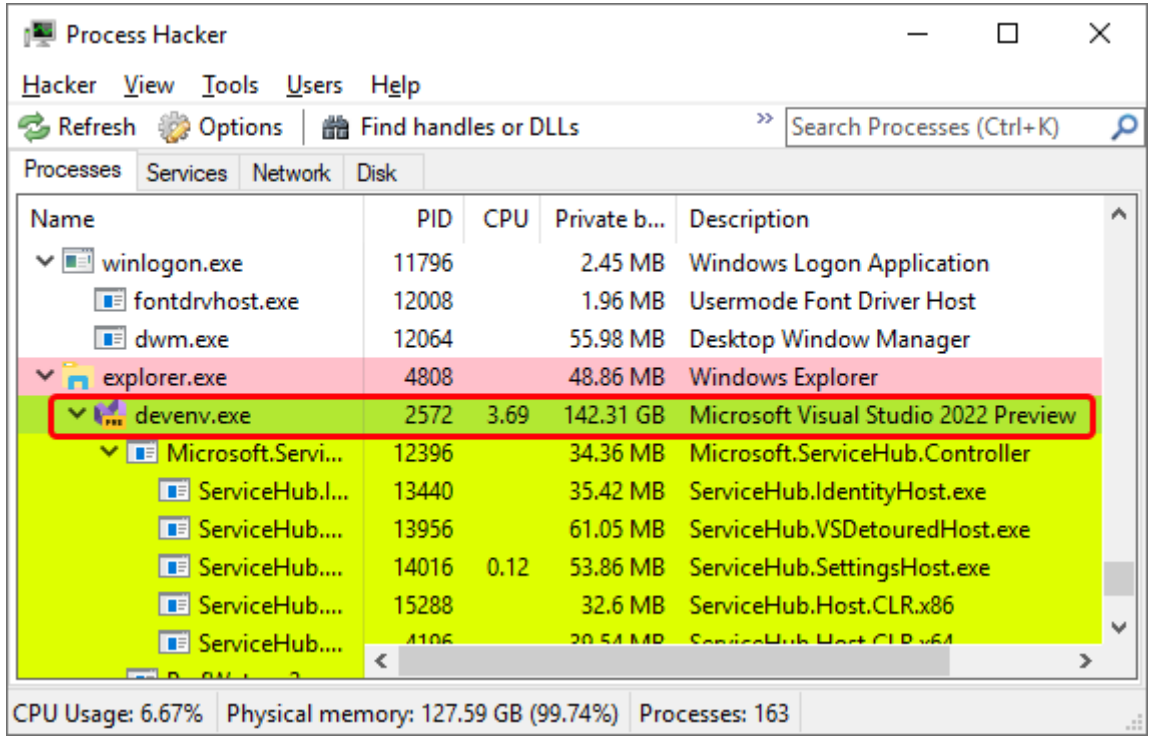
Тимлид PVS-Studio C#

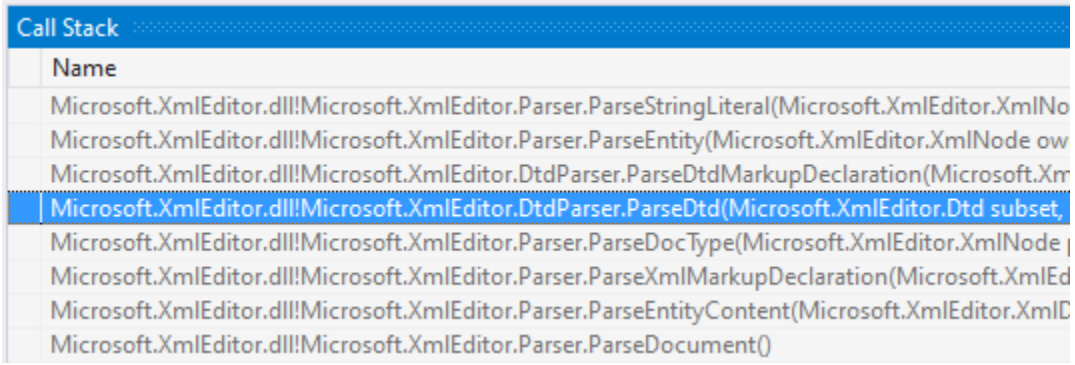
Пишу про .NET и C#




AUTOSAR
SAFETY
OWASP
CWE
AST
DAST
ASVS
MISRA
CVE
SAST
SECURITY
SCA


Task Manager
File Options View
Processes Performance Users Details Services
Name
7%
CPU
99%
Memory
Apps (3)
Microsoft Visual Studio 2022 Preview (8) 4.8% 124,621.9 MB
Microsoft.ServiceHub.Controller 0% 20.4 MB
PerfWatson2.exe 0% 42.9 MB
ServiceHub.Host.CLR.x64 0% 27.7 MB
ServiceHub.Host.CLR.x86 (32 bit) 0% 25.5 MB
ServiceHub.IdentityHost.exe (32 bit) 0% 27.4 MB
ServiceHub.SettingsHost.exe (32 bit) 0% 37.8 MB
ServiceHub.VSDetouredHost.exe 0% 36.0 MB
Microsoft Visual Studio Preview 4.8% 124,404.3 MB
Fewer details
End task

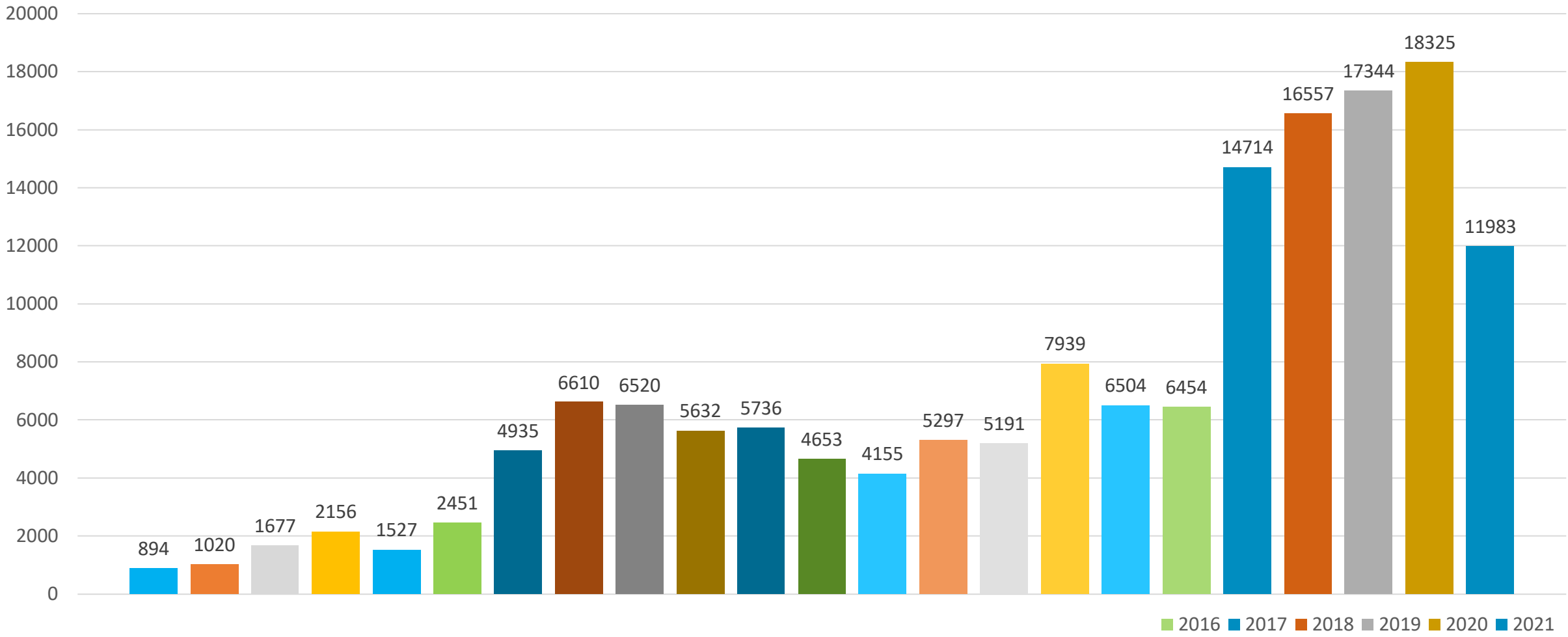

Process Hacker
Hacker View Tools Users Help
Refresh Options Find handles or DLLs
Search Processes (Ctrl+K)
Processes Services Network Disk
Name PID CPU Private b... Description
winlogon.exe 11796 2.45 MB Windows Logon Application
fontdrvhost.exe 12008 1.96 MB Usermode Font Driver Host
dwm.exe 12064 55.98 MB Desktop Window Manager
explorer.exe 4808 48.86 MB Windows Explorer
devenv.exe 2572 3.69 142.31 GB Microsoft Visual Studio 2022 Preview
Microsoft.Servi... 12396 34.36 MB Microsoft.ServiceHub.Controller
ServiceHub.I... 13440 35.42 MB ServiceHub.IdentityHost.exe
ServiceHub.... 13956 61.05 MB ServiceHub.VSDetouredHost.exe
ServiceHub.... 14016 0.12 53.86 MB ServiceHub.SettingsHost.exe
ServiceHub.... 15288 32.6 MB ServiceHub.Host.CLR.x86
ServiceHub....
CPU Usage: 6.67% Physical memory: 127.59 GB (99.74%) Processes: 163


Call Stack
Name
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseStringLiteral(Microsoft.XmlEditor.XmlNo
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseEntity(Microsoft.XmlEditor.XmlNode ow
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.DtdParser.ParseDtdMarkupDeclaration(Microsoft.Xm
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.DtdParser.ParseDtd(Microsoft.XmlEditor.Dtd subset,
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseDocType(Microsoft.XmlEditor.XmlNode
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseXmlMarkupDeclaration(Microsoft.XmlEd
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseEntityContent(Microsoft.XmlEditor.XmlD
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseDocument()


'; DROP TABLE PLATES;
S3
Audi - Vorsprung durch Technik

SAST

График уязвимостей по годам (1999-2021)



Баг или уязвимость?

```
$password = mysql_escape_string($_POST['password']);
....
if (     $password == 'astridservice'
     and $stilecustumization == 'astrid') {
   ....
}
```

CVE-2012-5862

```
$password = mysql_escape_string($_POST['password']);
....
if (     $password == 'astridservice'
     and $stilecustumization == 'astrid') {
   ....
}
```

CVE-2012-5862

```
$password = mysql_escape_string($_POST['password']);
....
if (     $password == 'astridservice'
     and $stilecustumization == 'astrid') {
   ....
}
```

```
typedef char my_bool;

my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}
```

CVE-2012-2122

```
typedef char my_bool;

my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}
```

CVE-2012-2122

```
typedef char my_bool;

my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}
```

CVE-2012-2122

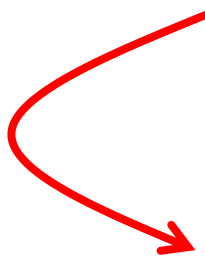
```
typedef char my_bool;
my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}
```

CVE-2012-2122

```
typedef char my_bool;
my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}
```

CVE-2012-2122

```
typedef char my_bool;
my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}       int
```



CVE-2012-2122

```
typedef char my_bool;
my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}
```

int -> char

Закрепление: баги и уязвимости

Терминология

Терминология

Что есть что?

Как связано?

Безопасность и защищённость

Безопасность и защищённость

- Safety (безопасность) / security (защищённость)

Безопасность и защищённость

- Safety (безопасность) / security (защищённость)
- Безопасность:
  - MISRA C
  - MISRA C++
  - AUTOSAR C++ Coding Guidelines

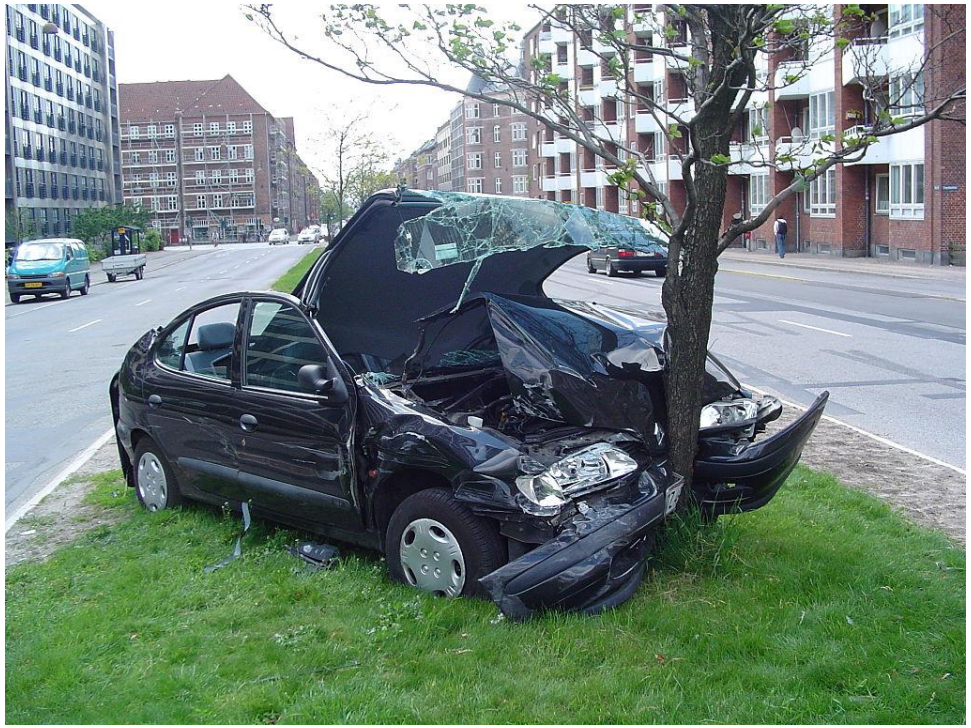
Безопасность и защищённость

- Safety (безопасность) / security (защищённость)
- Безопасность:
  - MISRA C
  - MISRA C++
  - AUTOSAR C++ Coding Guidelines
- Защищённость:
  - OWASP ASVS
  - OWASP Top 10

Безопасность

- Про надёжность
  (чтобы работало как швейцарские часы)
- Особенно актуальна там,
  где стоимость ошибки критична

- Должно надёжно работать без
  вмешательств извне



Защищённость

- Про конфиденциальные данные
- Про устойчивость к атакам

- Должно надёжно работать при попытках вмешательств извне

MISRA

MISRA

- MISRA: Motor Industry Software Reliability Association
  - MISRA C: 2012
  - MISRA C++: 2008

MISRA

- MISRA: Motor Industry Software Reliability Association
  - MISRA C: 2012
  - MISRA C++: 2008
- Посыл – максимально простой, надёжный, читаемый код

MISRA

- MISRA: Motor Industry Software Reliability Association
  - MISRA C: 2012
  - MISRA C++: 2008
- Посыл – максимально простой, надёжный, читаемый код

- Примеры правил:
  - не использовать goto
  - не использовать восьмеричные константы
  - не использовать динамическую память
  - все условные выражения должны быть с фигурными скобками

MISRA

```
void Foo(bool flag)
{
  if (flag)
    DoSmth();
}
```

MISRA

```
void Foo(bool flag)
{
  if (flag)
    DoSmth();
}
```

MISRA

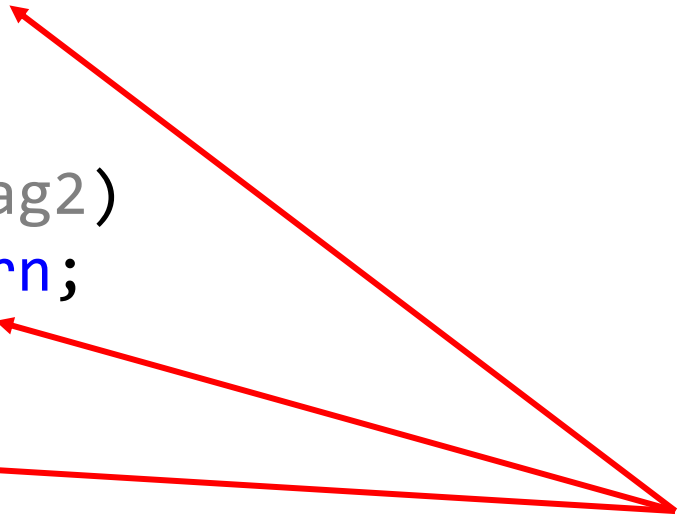
```
void Foo(bool flag)
{
  if (flag)
  {
    DoSmth();
  }
}
```

MISRA

```
void Foo(bool flag1, bool flag2)
{
  ....
  if (flag1)
    return;

  ....
  if (flag2)
    return;

  ....
}
```

MISRA

```
void Foo(bool flag1, bool flag2)
{
  ....
  if (flag1)
    return;

  ....
  if (flag2)
    return;

  ....
}
```



exit points
(> 1)

Закрепление: безопасность

- Стандарты важны, если пишете критичный к безопасности софт

- Примеры:
  - MISRA C
  - MISRA C++
  - AUTOSAR (AUTomotive Open System ARchitecture) C++ Coding Guidelines

Потенциальные уязвимости (CWE)

CWE

- CWE: Common Weakness Enumeration
- Паттерны описания потенциальных уязвимостей
- Альт. "недостаток безопасности"

```
int id_sequence[3];

/* Populate the id array. */

id_sequence[0] = 123;
id_sequence[1] = 234;
id_sequence[2] = 345;
id_sequence[3] = 456;
```

```
int id_sequence[3];

/* Populate the id array. */

id_sequence[0] = 123;
id_sequence[1] = 234;
id_sequence[2] = 345;
id_sequence[3] = 456;
```

CWE-787: Out-of-bounds Write

```
int id_sequence[3];

/* Populate the id array. */

id_sequence[0] = 123;
id_sequence[1] = 234;
id_sequence[2] = 345;
id_sequence[3] = 456;
```

CWE-787: Out-of-bounds Write

Weakness ID: 787
Abstraction: Base
Structure: Simple

Status: Draft

Presentation Filter: Complete

Description

The software writes data past the end, or before the beginning, of the intended buffer.

Extended Description

Typically, this can result in corruption of data, a crash, or code execution. The software may modify an index or perform pointer arithmetic that references a memory location that is outside of the boundaries of the buffer. A subsequent write operation then produces undefined or unexpected results.

Alternate Terms

Memory Corruption: The generic term "memory corruption" is often used to describe the consequences of writing to memory outside the bounds of a buffer, or to memory that is invalid, when the root cause is something other than a sequential copy of excessive data from a fixed starting location. This may include issues such as incorrect pointer arithmetic, accessing invalid pointers due to incomplete initialization or memory release, etc.

Relationships

Modes Of Introduction

Applicable Platforms

Common Consequences

Likelihood Of Exploit

Demonstrative Examples

Example 1

The following code attempts to save four different identification numbers into an array.

Example Language: **C** *(bad code)*

```
int id_sequence[3];

/* Populate the id array. */

id_sequence[0] = 123;
id_sequence[1] = 234;
id_sequence[2] = 345;
id_sequence[3] = 456;
```

Since the array is only allocated to hold three elements, the valid indices are 0 to 2; so, the assignment to id_sequence[3] is out of bounds.

Example 2

In the following example, it is possible to request that memcpy move a much larger segment of memory than assumed:

Example Language: **C** *(bad code)*

```
int returnChunkSize(void *) {

  /* if chunk info is valid, return the size of usable memory,

   * else, return -1 to indicate an error

   */
```

```
void GetData(char* MFAddr)
{
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd)))
  {
    if (ConnectToMainframe(MFAddr, pwd))
    {
      // Interaction with mainframe
    }
  }

  memset(pwd, 0, sizeof(pwd));
}
```

```
void GetData(char* MFAddr)
{
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd)))
  {
    if (ConnectToMainframe(MFAddr, pwd))
    {
      // Interaction with mainframe
    }
  }

  memset(pwd, 0, sizeof(pwd));
}
```

```
void GetData(char* MFAddr)
{
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd)))
  {
    if (ConnectToMainframe(MFAddr, pwd))
    {
      // Interaction with mainframe
    }
  }

  memset(pwd, 0, sizeof(pwd));
}
```

```
void GetData(char* MFAddr)
{
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd)))
  {
    if (ConnectToMainframe(MFAddr, pwd))
    {
      // Interaction with mainframe
    }
  }

  memset(pwd, 0, sizeof(pwd));
}
```

```
void GetData(char* MFAddr)
{
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd)))
  {
    if (ConnectToMainframe(MFAddr, pwd))
    {
      // Interaction with mainframe
    }
  }

  memset(pwd, 0, sizeof(pwd));
}
```

```
void GetData(char* MFAddr)
{
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd)))
  {
    if (ConnectToMainframe(MFAddr, pwd))
    {
      // Interaction with mainframe
    }
  }

  memset(pwd, 0, sizeof(pwd));
}
```

CWE-14:
Compiler Removal of Code to Clear Buffers

```
void GetData(char* MFAddr)
{
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd)))
  {
    if (ConnectToMainframe(MFAddr, pwd))
    {
      // Interaction with mainframe
    }
  }

  memset(pwd, 0, sizeof(pwd));
}
```

CWE-14: Compiler Removal of Code to Clear Buffers

Weakness ID: 14
Abstraction: Variant
Structure: Simple

Status: Draft

Presentation Filter: Complete

Description

Sensitive memory is cleared according to the source code, but compiler optimizations leave the memory untouched when it is not read from again, aka "dead store removal."

Extended Description

This compiler optimization error occurs when:

- 1. Secret data are stored in memory.
- 2. The secret data are scrubbed from memory by overwriting its contents.
- 3. The source code is compiled using an optimizing compiler, which identifies and removes the function that overwrites the contents as a dead store because the memory is not used subsequently.

Relationships

Modes Of Introduction

Applicable Platforms

Common Consequences

Demonstrative Examples

Example 1

The following code reads a password from the user, uses the password to connect to a back-end mainframe and then attempts to scrub the password from memory using memset().

Example Language: **C** *(bad code)*

```
void GetData(char *MFAddr) {
  char pwd[64];
  if (GetPasswordFromUser(pwd, sizeof(pwd))) {

    if (ConnectToMainframe(MFAddr, pwd)) {

      // Interaction with mainframe
    }
  }
  memset(pwd, 0, sizeof(pwd));
}
```

The code in the example will behave correctly if it is executed verbatim, but if the code is compiled using an optimizing compiler, such as Microsoft Visual C++ .NET or GCC 3.x, then the call to memset() will be removed as a dead store because the buffer pwd is not used after its value is overwritten [18]. Because the buffer pwd contains a sensitive value, the application may be vulnerable to attack if the data are left memory resident. If attackers are able to access the correct region of memory, they may use the recovered password to gain control of the system.

It is common practice to overwrite sensitive data manipulated in memory, such as passwords or cryptographic keys, in order to prevent attackers from learning system secrets. However, with the advent of optimizing compilers, programs do not always behave as their source code alone would suggest. In the example, the compiler interprets the call to memset() as dead code because the memory being written to is not subsequently used, despite the fact that there is clearly a security motivation for the operation to occur. The problem here is that many compilers, and in fact many programming languages, do not take this and other security concerns into consideration in their efforts to improve efficiency.

Attackers typically exploit this type of vulnerability by using a core dump or runtime mechanism to access the memory used by a particular application and recover the secret information. Once an attacker has access to the secret information, it is relatively straightforward to further exploit the system and possibly compromise other resources with which the application interacts.

Potential Mitigations

Закрепление: CWE

- Классификация недостатков безопасности
- Потенциальные уязвимости

Закрепление: CWE

- Классификация недостатков безопасности
- Потенциальные уязвимости

- CWE Top 25 Most Dangerous Software Weaknesses

Rank	ID	Name	Score	2020 Rank Change
[1]	CWE-787	Out-of-bounds Write	65.93	+1
[2]	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	46.84	-1
[3]	CWE-125	Out-of-bounds Read	24.9	+1
[4]	CWE-20	Improper Input Validation	20.47	-1
[5]	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	19.55	+5
[6]	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	19.54	0
[7]	CWE-416	Use After Free	16.83	+1
[8]	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	14.69	+4
[9]	CWE-352	Cross-Site Request Forgery (CSRF)	14.46	0
[10]	CWE-434	Unrestricted Upload of File with Dangerous Type	8.45	+5
[11]	CWE-306	Missing Authentication for Critical Function	7.93	+13
[12]	CWE-190	Integer Overflow or Wraparound	7.12	-1
[13]	CWE-502	Deserialization of Untrusted Data	6.71	+8
[14]	CWE-287	Improper Authentication	6.58	0
[15]	CWE-476	NULL Pointer Dereference	6.54	-2

Уязвимости (CVE)

Уязвимости (CVE)

- CVE: Common Vulnerabilities and Exposures
- Запись из базы CVE описывает не теоретическую опасность, а конкретную уязвимость в приложении

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
   ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
   ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

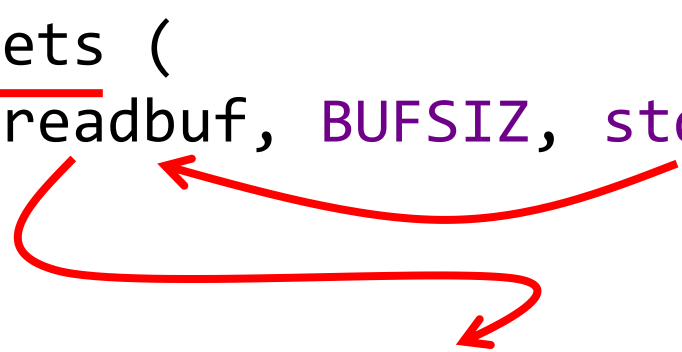
libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
   ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```



Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
   ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

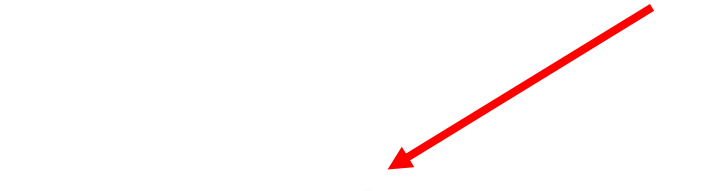
Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
   ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```



Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
   ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

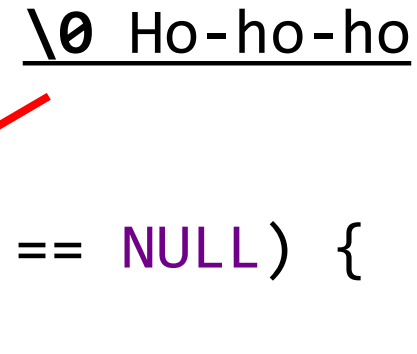
libidn

\0 Ho-ho-ho

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
   ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

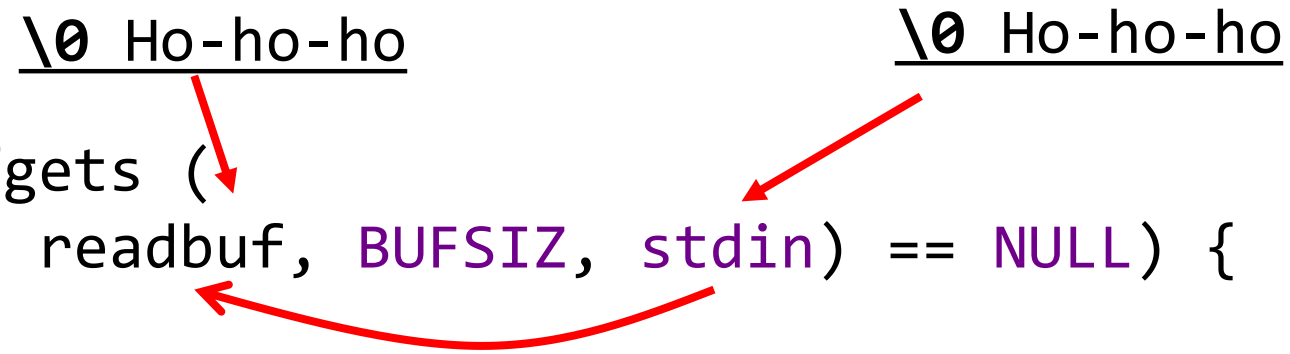
libidn




```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

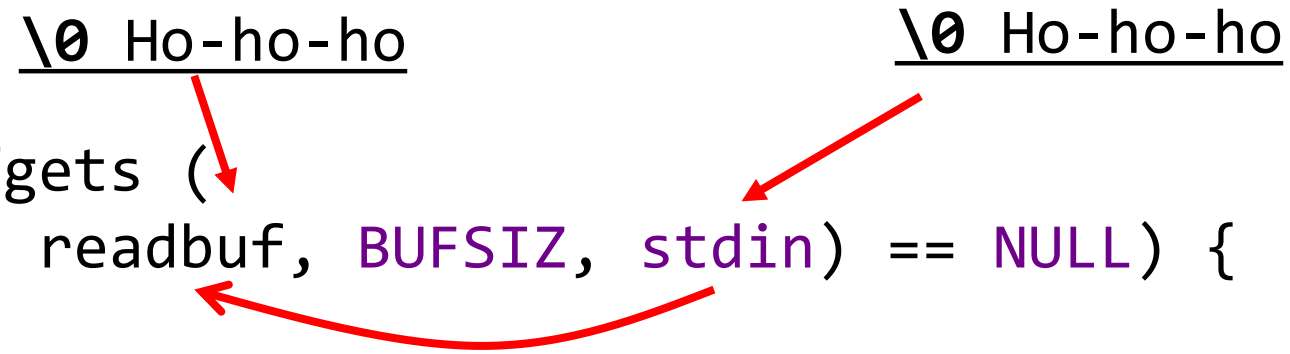
libidn




```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

Common Vulnerabilities and Exposures

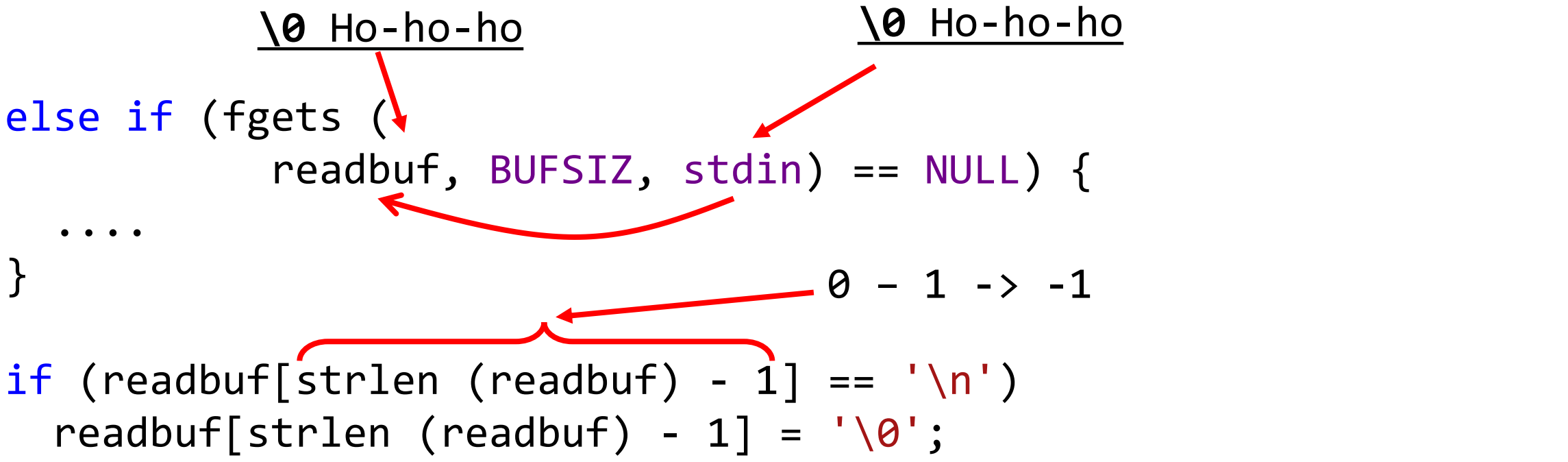
libidn




```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

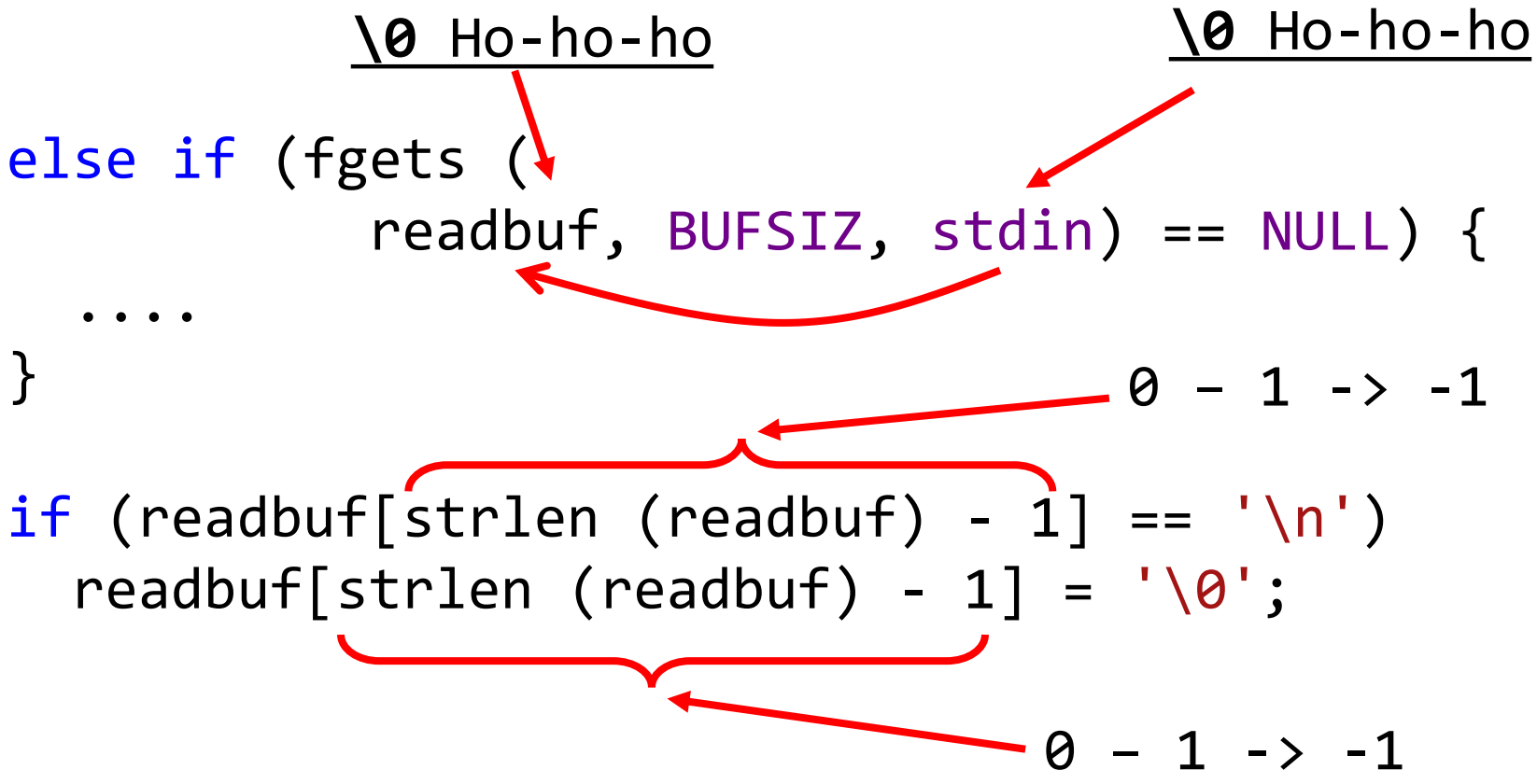
Common Vulnerabilities and Exposures

libidn



Common Vulnerabilities and Exposures

libidn



Common Vulnerabilities and Exposures

CVE-2015-8948

CVE-ID

CVE-2015-8948 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

idn in GNU libidn before 1.33 might allow remote attackers to obtain sensitive memory information by reading a zero byte as input, which triggers an out-of-bounds read.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- BID:92070
- URL:http://www.securityfocus.com/bid/92070
- CONFIRM:http://git.savannah.gnu.org/cgit/libidn.git/commit/?id=570e68886c41c2e765e6218cb317d9a9a447a041
- DEBIAN:DSA-3658
- URL:http://www.debian.org/security/2016/dsa-3658
- MLIST:[bookkeeper-issues] 20210628 [GitHub] [bookkeeper] padma81 opened a new issue #2746: Security Vulnerabilities in CentOS 7 image, Upgrade image to CentOS 8
- URL:https://lists.apache.org/thread.html/rf4c02775860db415b4955778a131c2795223f61cb8c6a450893651e4@%3Cissues.bookkeeper.apache.org%3E
- MLIST:[bookkeeper-issues] 20210629 [GitHub] [bookkeeper] padma81 opened a new issue #2746: Security Vulnerabilities in CentOS 7 image, Upgrade image to CentOS 8
- URL:https://lists.apache.org/thread.html/r58af02e294bd07f487e2c64ffc0a29b837db5600e33b6e698b9d696b@%3Cissues.bookkeeper.apache.org%3E
- MLIST:[help-libidn] 20160720 Libidn 1.33 released
- URL:https://lists.gnu.org/archive/html/help-libidn/2016-07/msg00009.html
- MLIST:[oss-security] 20160720 CVE request: multiple issues fixed in GNU libidn 1.33
- URL:http://www.openwall.com/lists/oss-security/2016/07/20/6
- MLIST:[oss-security] 20160721 Re: CVE request: multiple issues fixed in GNU libidn 1.33
- URL:http://www.openwall.com/lists/oss-security/2016/07/21/4
- SUSE:openSUSE-SU-2016:1924
- URL:http://lists.opensuse.org/opensuse-updates/2016-08/msg00005.html
- SUSE:openSUSE-SU-2016:2135
- URL:http://lists.opensuse.org/opensuse-updates/2016-08/msg00098.html
- UBUNTU:USN-3068-1
- URL:http://www.ubuntu.com/usn/USN-3068-1

Assigning CNA

MITRE Corporation

Date Record Created

20160721 Disclaimer: The record creation date may reflect when the CVE ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Assigned (20160721)

Votes (Legacy)

Comments (Legacy)

Proposed (Legacy)

N/A

This is a record on the CVE List, which provides common identifiers for publicly known cybersecurity vulnerabilities.

SEARCH CVE USING KEYWORDS: Submit
You can also search by reference using the CVE Reference Maps.

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

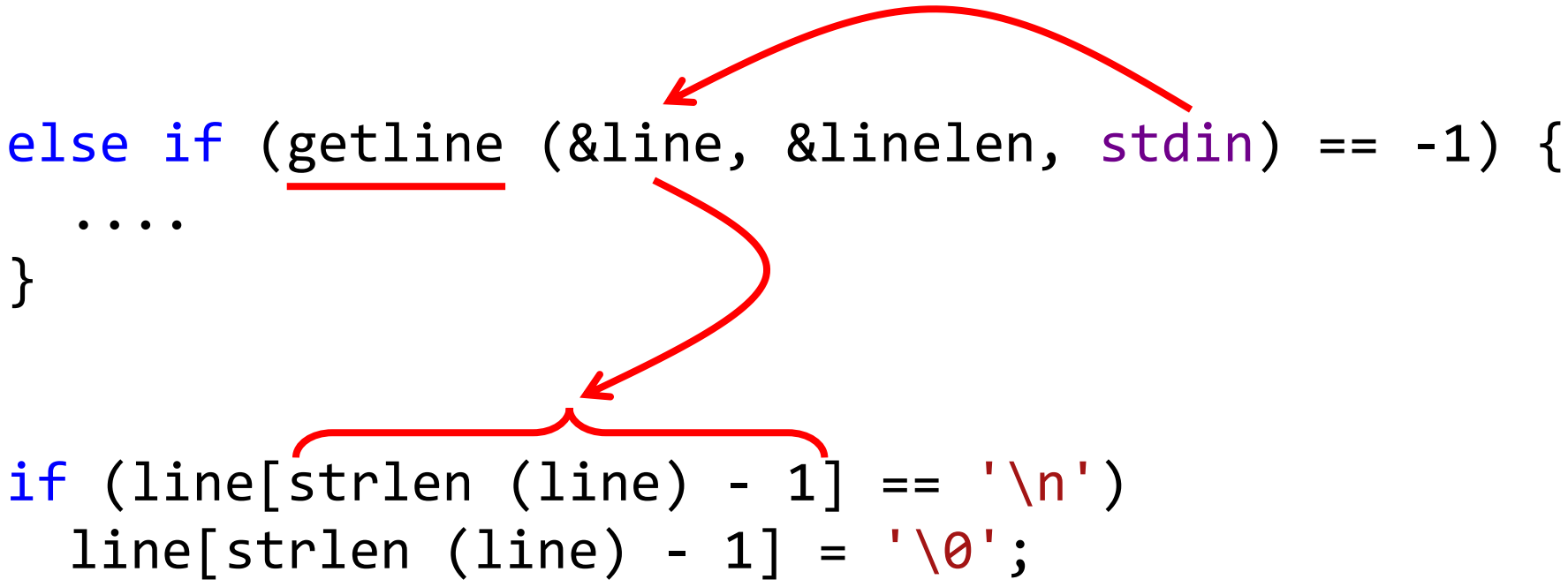
Common Vulnerabilities and Exposures

libidn

```
else if (getline (&line, &linelen, stdin) == -1) {
   ....
}
```

```
if (line[strlen (line) - 1] == '\n')
  line[strlen (line) - 1] = '\0';
```

libidn



```
else if (getline (&line, &linelen, stdin) == -1) {
   ....
}


if (line[strlen (line) - 1] == '\n')
  line[strlen (line) - 1] = '\0';
```

Common Vulnerabilities and Exposures

CVE-2016-6262

CVE-ID

CVE-2016-6262 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

idn in libidn before 1.33 might allow remote attackers to obtain sensitive memory information by reading a zero byte as input, which triggers an out-of-bounds read, a different vulnerability than CVE-2015-8948.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- BID:92070
- URL:http://www.securityfocus.com/bid/92070
- CONFIRM:http://git.savannah.gnu.org/cgit/libidn.git/commit/?id=5e3cb9c7b5bf0ce665b9d68f5ddf095af5c9ba60
- MLIST:[bookkeeper-issues] 20210628 [GitHub] [bookkeeper] padma81 opened a new issue #2746: Security Vulnerabilities in CentOS 7 image, Upgrade image to CentOS 8
- URL:https://lists.apache.org/thread.html/rf4c02775860db415b4955778a131c2795223f61cb8c6a450893651e4@%3Cissues.bookkeeper.apache.org%3E
- MLIST:[bookkeeper-issues] 20210629 [GitHub] [bookkeeper] padma81 opened a new issue #2746: Security Vulnerabilities in CentOS 7 image, Upgrade image to CentOS 8
- URL:https://lists.apache.org/thread.html/r58af02e294bd07f487e2c64ffc0a29b837db5600e33b6e698b9d696b@%3Cissues.bookkeeper.apache.org%3E
- MLIST:[help-libidn] 20160720 Libidn 1.33 released
- URL:https://lists.gnu.org/archive/html/help-libidn/2016-07/msg00009.html
- MLIST:[oss-security] 20160720 CVE request: multiple issues fixed in GNU libidn 1.33
- URL:http://www.openwall.com/lists/oss-security/2016/07/20/6
- MLIST:[oss-security] 20160721 Re: CVE request: multiple issues fixed in GNU libidn 1.33
- URL:http://www.openwall.com/lists/oss-security/2016/07/21/4
- SUSE:openSUSE-SU-2016:1924
- URL:http://lists.opensuse.org/opensuse-updates/2016-08/msg00005.html
- SUSE:openSUSE-SU-2016:2135
- URL:http://lists.opensuse.org/opensuse-updates/2016-08/msg00098.html
- UBUNTU:USN-3068-1
- URL:http://www.ubuntu.com/usn/USN-3068-1

Assigning CNA

MITRE Corporation

Date Record Created

20160721 Disclaimer: The record creation date may reflect when the CVE ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Assigned (20160721)

Votes (Legacy)

Comments (Legacy)

Proposed (Legacy)

N/A

This is a record on the CVE List, which provides common identifiers for publicly known cybersecurity vulnerabilities.

SEARCH CVE USING KEYWORDS: Submit
You can also search by reference using the CVE Reference Maps.

For More Information: CVE Request Web Form (select "Other" from dropdown)

libidn

```
else if (getline (&line, &linelen, stdin) == -1) {
   ....
}
```

```
if (line[strlen (line) - 1] == '\n')
  line[strlen (line) - 1] = '\0';
```

libidn

```
else if (getline (&line, &linelen, stdin) == -1) {
   ....
}

if (strlen (line) > 0)
  if (line[strlen (line) - 1] == '\n')
    line[strlen (line) - 1] = '\0';
```

libidn

```
else if (getline (&line, &linelen, stdin) == -1) {
  ....
}

if (strlen (line) > 0)
  if (line[strlen (line) - 1] == '\n')
    line[strlen (line) - 1] = '\0';
```

Common Vulnerabilities and Exposures

CVE из libidn

CVE-2015-8948

Коммит, "закрывающий" уязвимость: *10.08.2015*

CVE-2016-6262

Коммит, закрывающий уязвимость: *14.01.2016*

Разница – *5 месяцев*

Закрепление: CVE

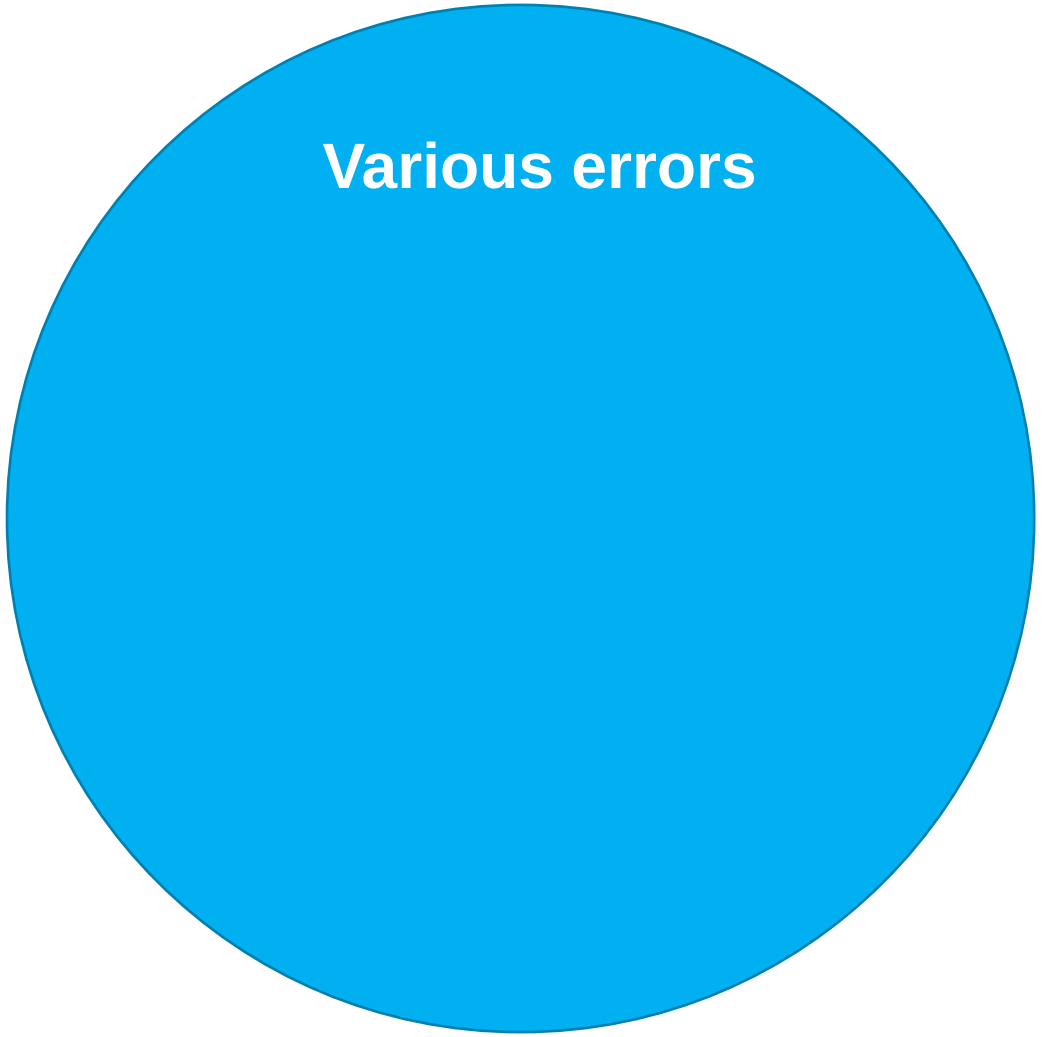
- Реальные уязвимости
- CWE описывает паттерны,
  CVE - их конкретные проявления

Закрепление: CVE

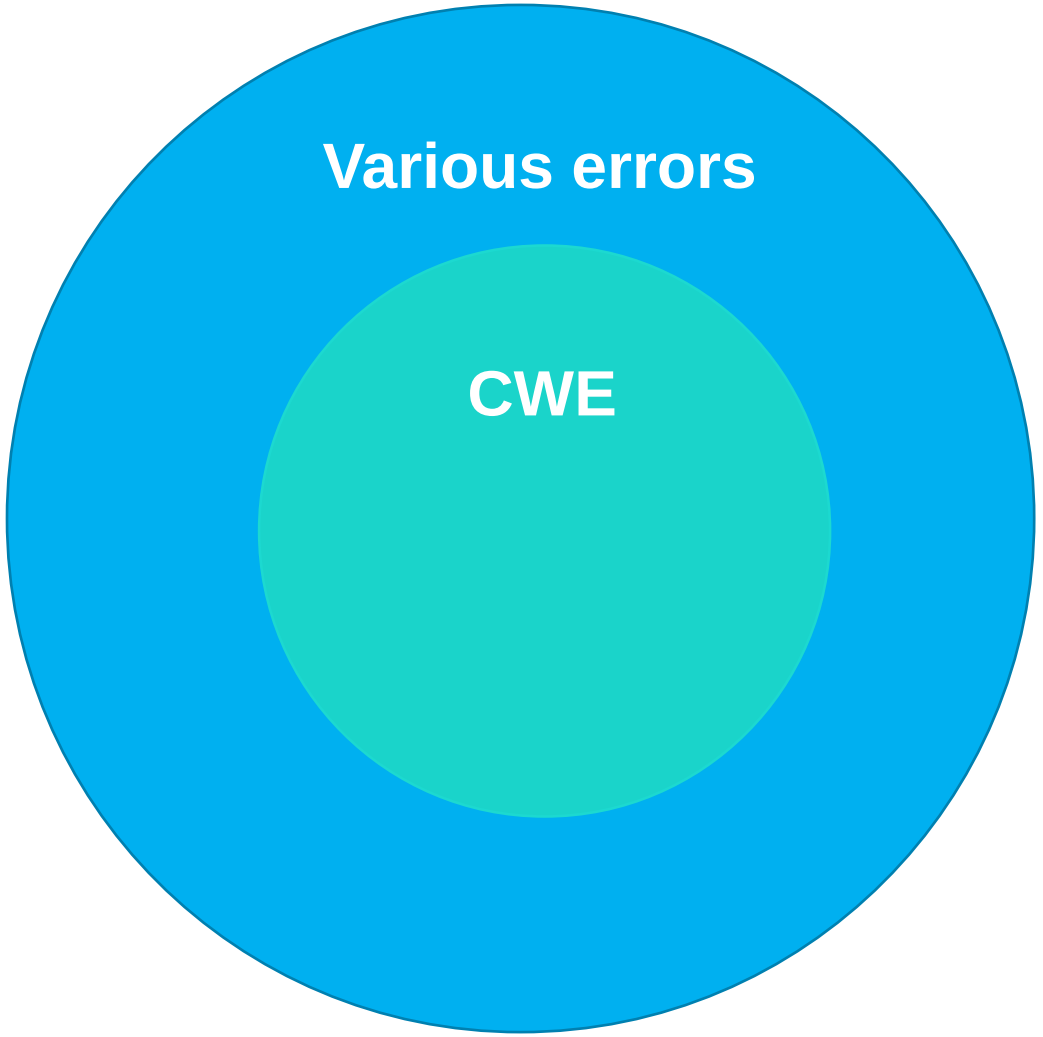
- Реальные уязвимости
- CWE описывает паттерны,
  CVE - их конкретные проявления
  (в частности)

Связь багов, CWE и CVE

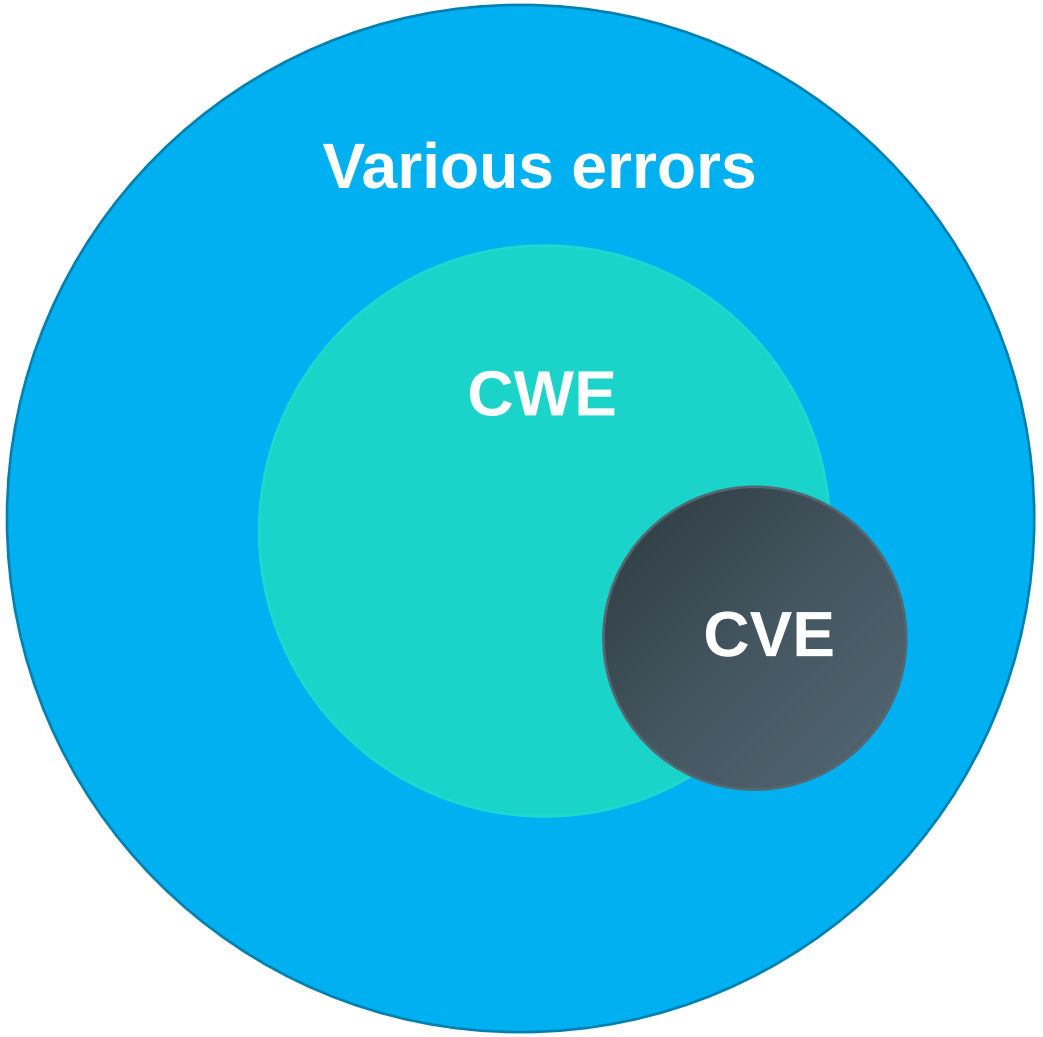
Связь багов, CWE и CVE



Связь багов, CWE и CVE



Связь багов, CWE и CVE



OWASP

OWASP

- OWASP: Open Web Application Security Project
- OWASP ASVS: OWASP Application Security Verification Standard
- OWASP Top 10

OWASP Top 10 2017

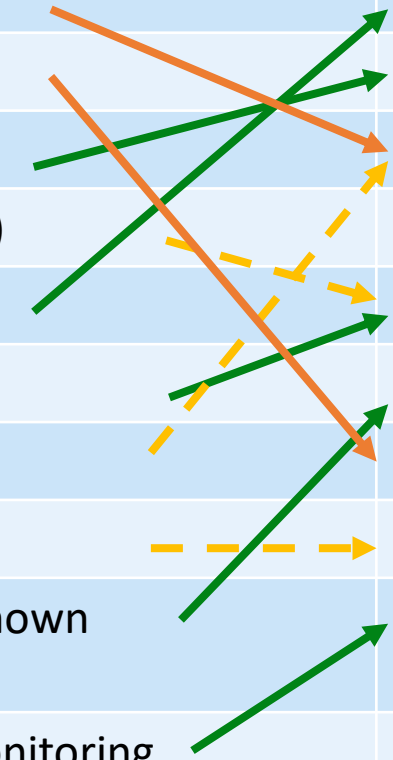
- A1:2017 - Injection
- A2:2017 - Broken Authentication
- A3:2017 - Sensitive Data Exposure
- A4:2017 - XML External Entities (XXE)
- A5:2017 - Broken Access Control
- A6:2017 - Security Misconfiguration
- A7:2017 - Cross-Site Scripting (XSS)
- A8:2017 - Insecure Deserialization
- A9:2017 - Using Components with Known Vulnerabilities
- A10:2017 - Insufficient Logging & Monitoring

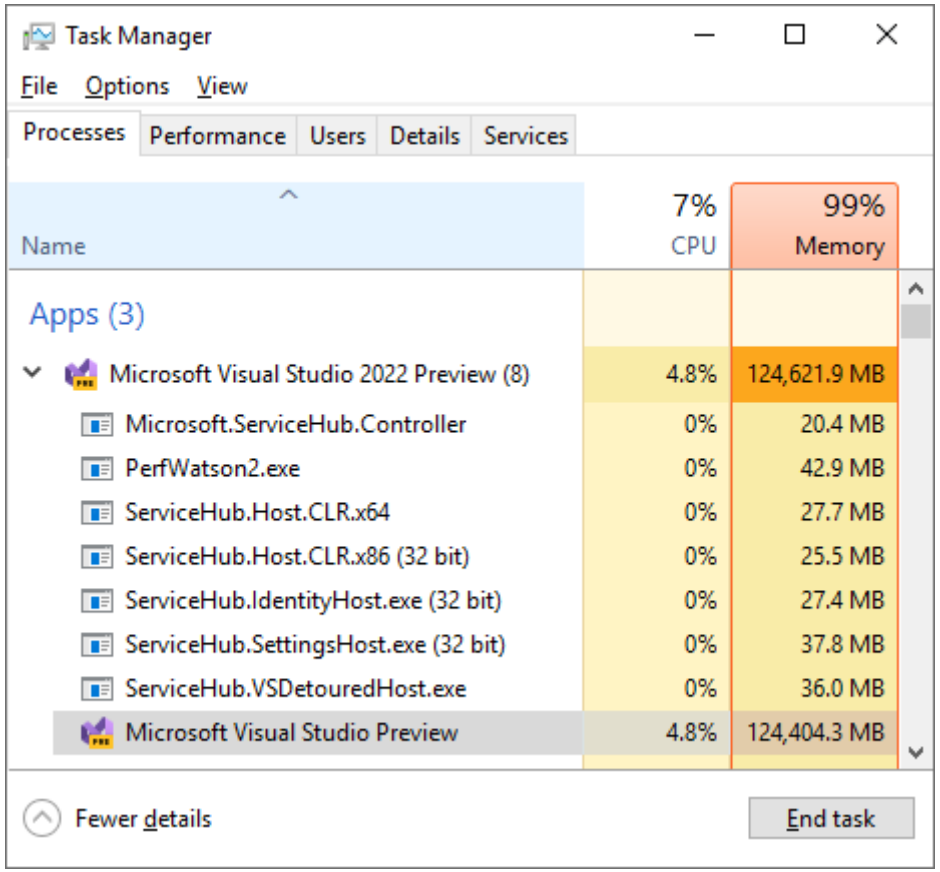
OWASP Top 10 2017 vs 2021

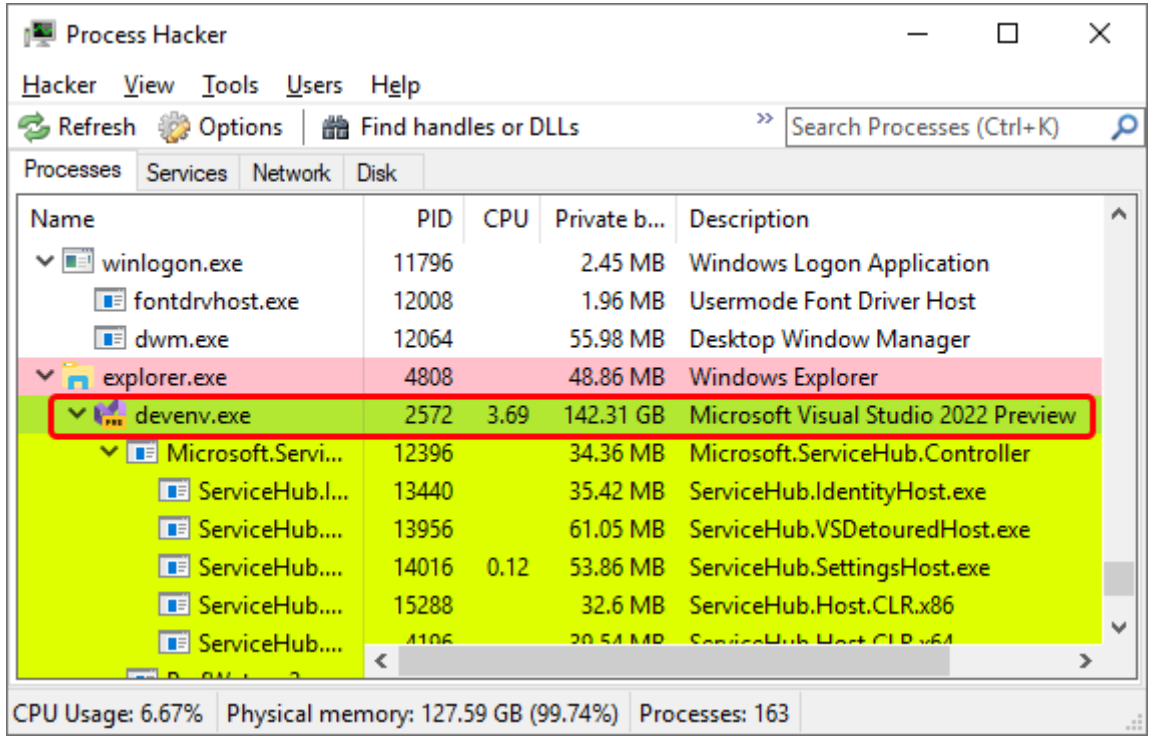
OWASP Top 10 2017	OWASP Top 10 2021
A1:2017 - Injection	A01:2021 - Broken Access Control
A2:2017 - Broken Authentication	A02:2021 - Cryptographic Failures
A3:2017 - Sensitive Data Exposure	A03:2021 – Injection
A4:2017 - XML External Entities (XXE)	A04:2021 - Insecure Design
A5:2017 - Broken Access Control	A05:2021 - Security Misconfiguration
A6:2017 - Security Misconfiguration	A06:2021 - Vulnerable and Outdated Components
A7:2017 - Cross-Site Scripting (XSS)	A07:2021 - Identification and Authentication Failures
A8:2017 - Insecure Deserialization	A08:2021 - Software and Data Integrity Failures
A9:2017 - Using Components with Known Vulnerabilities	A09:2021 - Security Logging and Monitoring Failures
A10:2017 - Insufficient Logging & Monitoring	A10:2021 - Server-Side Request Forgery

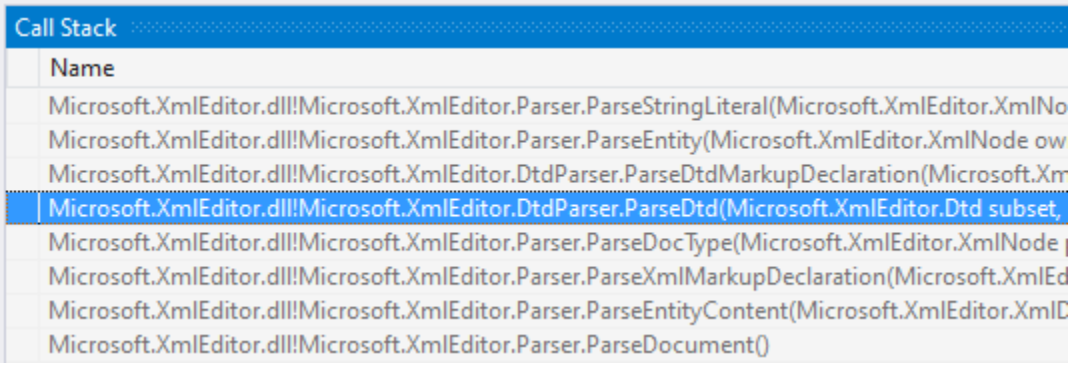
OWASP Top 10 2017 vs 2021

OWASP Top 10 2017	OWASP Top 10 2021
A1:2017 - Injection	A01:2021 - Broken Access Control
A2:2017 - Broken Authentication	A02:2021 - Cryptographic Failures
A3:2017 - Sensitive Data Exposure	A03:2021 – Injection
A4:2017 - XML External Entities (XXE)	A04:2021 - Insecure Design
A5:2017 - Broken Access Control	A05:2021 - Security Misconfiguration
A6:2017 - Security Misconfiguration	A06:2021 - Vulnerable and Outdated Components
A7:2017 - Cross-Site Scripting (XSS)	A07:2021 - Identification and Authentication Failures
A8:2017 - Insecure Deserialization	A08:2021 - Software and Data Integrity Failures
A9:2017 - Using Components with Known Vulnerabilities	A09:2021 - Security Logging and Monitoring Failures
A10:2017 - Insufficient Logging & Monitoring	A10:2021 - Server-Side Request Forgery




Task Manager
File Options View
Processes Performance Users Details Services
Name
7%
CPU
99%
Memory
Apps (3)
Microsoft Visual Studio 2022 Preview (8) 4.8% 124,621.9 MB
Microsoft.ServiceHub.Controller 0% 20.4 MB
PerfWatson2.exe 0% 42.9 MB
ServiceHub.Host.CLR.x64 0% 27.7 MB
ServiceHub.Host.CLR.x86 (32 bit) 0% 25.5 MB
ServiceHub.IdentityHost.exe (32 bit) 0% 27.4 MB
ServiceHub.SettingsHost.exe (32 bit) 0% 37.8 MB
ServiceHub.VSDetouredHost.exe 0% 36.0 MB
Microsoft Visual Studio Preview 4.8% 124,404.3 MB
Fewer details
End task


Process Hacker
Hacker View Tools Users Help
Refresh Options Find handles or DLLs
Search Processes (Ctrl+K)
Processes Services Network Disk
Name PID CPU Private b... Description
winlogon.exe 11796 2.45 MB Windows Logon Application
fontdrvhost.exe 12008 1.96 MB Usermode Font Driver Host
dwm.exe 12064 55.98 MB Desktop Window Manager
explorer.exe 4808 48.86 MB Windows Explorer
devenv.exe 2572 3.69 142.31 GB Microsoft Visual Studio 2022 Preview
Microsoft.Servi... 12396 34.36 MB Microsoft.ServiceHub.Controller
ServiceHub.I... 13440 35.42 MB ServiceHub.IdentityHost.exe
ServiceHub.... 13956 61.05 MB ServiceHub.VSDetouredHost.exe
ServiceHub.... 14016 0.12 53.86 MB ServiceHub.SettingsHost.exe
ServiceHub.... 15288 32.6 MB ServiceHub.Host.CLR.x86
ServiceHub....
CPU Usage: 6.67% Physical memory: 127.59 GB (99.74%) Processes: 163


Call Stack
Name
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseStringLiteral(Microsoft.XmlEditor.XmlNo
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseEntity(Microsoft.XmlEditor.XmlNode ow
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.DtdParser.ParseDtdMarkupDeclaration(Microsoft.Xm
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.DtdParser.ParseDtd(Microsoft.XmlEditor.Dtd subset,
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseDocType(Microsoft.XmlEditor.XmlNode
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseXmlMarkupDeclaration(Microsoft.XmlEd
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseEntityContent(Microsoft.XmlEditor.XmlD
Microsoft.XmlEditor.dll!Microsoft.XmlEditor.Parser.ParseDocument()

OWASP Top 10 2017

- A1:2017 - Injection
- A2:2017 - Broken Authentication
- A3:2017 - Sensitive Data Exposure
- A4:2017 - XML External Entities (XXE)
- A5:2017 - Broken Access Control
- A6:2017 - Security Misconfiguration
- A7:2017 - Cross-Site Scripting (XSS)
- A8:2017 - Insecure Deserialization
- A9:2017 - Using Components with Known Vulnerabilities
- A10:2017 - Insufficient Logging & Monitoring

A4:2017 - XML External Entities (XXE)

CWE CATEGORY: OWASP Top Ten 2017 Category A4 - XML External Entities (XXE)

Category ID: 1030

Summary

Weaknesses in this category are related to the A4 category in the OWASP Top Ten 2017.

Membership

Nature	Type	ID	Name
MemberOf	V	1026	Weaknesses in OWASP Top Ten (2017)
HasMember	B	611	Improper Restriction of XML External Entity Reference
HasMember	B	776	Improper Restriction of Recursive Entity References in DTDs ('XML Entity Expansion')

A4:2017 - XML External Entities (XXE)

CWE CATEGORY: OWASP Top Ten 2017 Category A4 - XML External Entities (XXE)

Category ID: 1030

Summary

Weaknesses in this category are related to the A4 category in the OWASP Top Ten 2017.

Membership

Nature	Type	ID	Name
MemberOf	V	1026	Weaknesses in OWASP Top Ten (2017)
HasMember	B	611	Improper Restriction of XML External Entity Reference
HasMember	B	776	Improper Restriction of Recursive Entity References in DTDs ('XML Entity Expansion')

A4:2017 - XML External Entities (XXE)

```
<!ENTITY myEntity "Entity value">
```

A4:2017 - XML External Entities (XXE)

```
<!ENTITY myEntity "Entity value">
....
<foo>&myEntity;</foo>
```

A4:2017 - XML External Entities (XXE)

```
<!ENTITY myEntity "Entity value">
....
<foo>&myEntity;</foo>


// -> <foo>Entity value</foo>
```

A4:2017 - XML External Entities (XXE)

```
<!ENTITY lol "lol">

<!ENTITY lol1
"&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
```

A4:2017 - XML External Entities (XXE)

```
<!ENTITY lol "lol">

<!ENTITY lol1
"&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">

// -> lollollollollollollollollollollol
```

A4:2017 - XML External Entities (XXE)

```
<!ENTITY lol2
"&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;
 &lol1;">

// ->
lollollollollollollollollollollollollollollollollollollol
lollollollollollollollollollollollollollollollollollollol
lollollollollollollollollollollollollollollollollollollol
lollollollollollollollollollollollollollollollollollollol
lollollollollollollollollollollol...
```

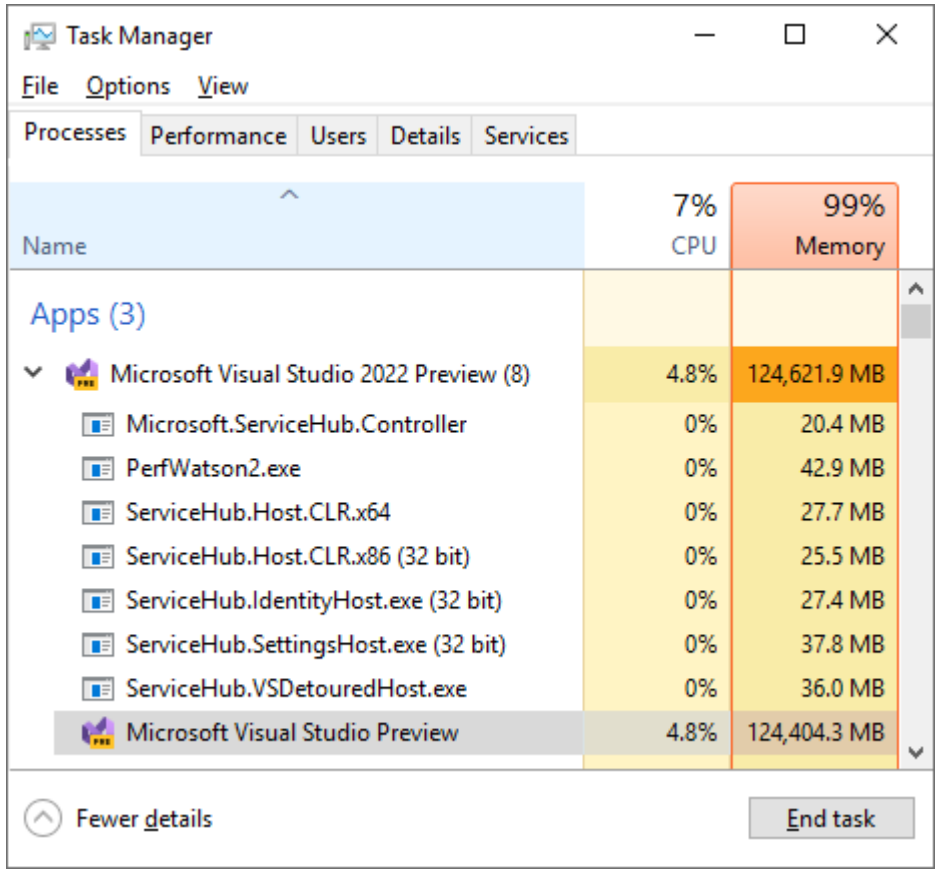
A4:2017 - XML External Entities (XXE)

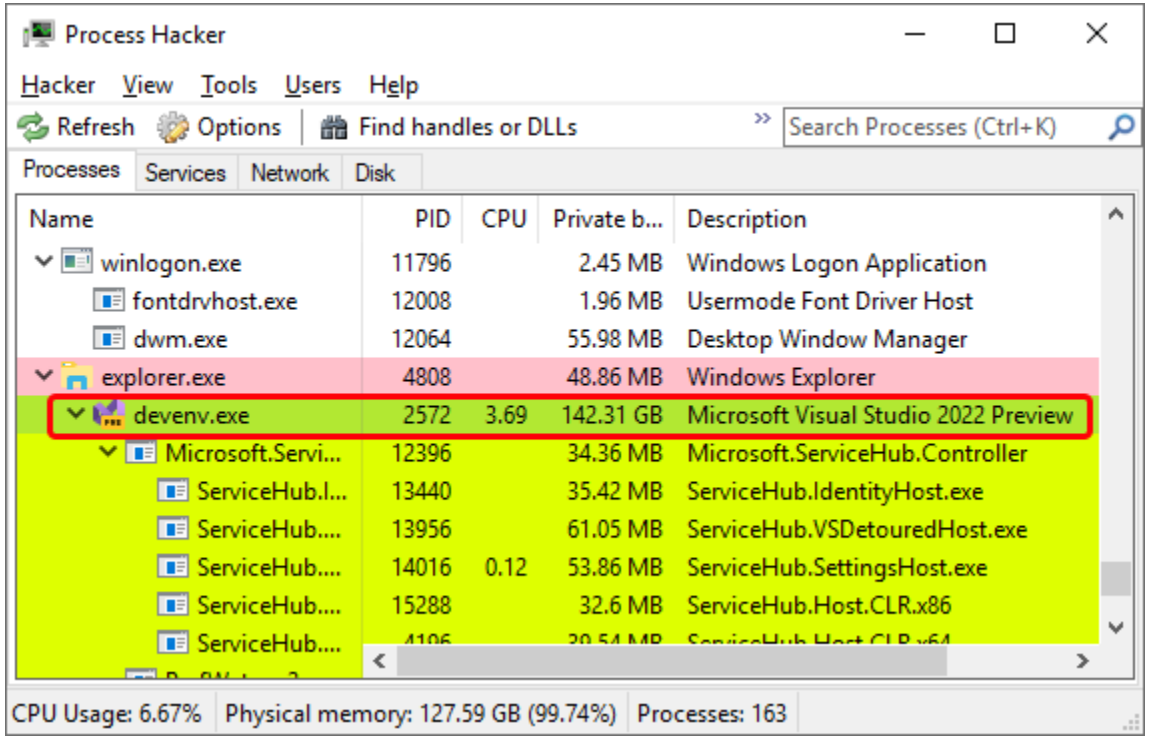
```
<?xml version="1.0"?>
<!DOCTYPE lolz
[
  <!ENTITY lol "lol">
  <!ELEMENT lolz (#PCDATA)>
  <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
  <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
  <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
  <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
  <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
  <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
  <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
  <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
  <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
  <!ENTITY lol10 "&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;">
  <!ENTITY lol11 "&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;">
  <!ENTITY lol12 "&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;">
  <!ENTITY lol13 "&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;">
  <!ENTITY lol14 "&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;">
  <!ENTITY lol15 "&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;">
]>
<lolz>&lol15;</lolz>
```

A4:2017 - XML External Entities (XXE)

```
<?xml version="1.0"?>
<!DOCTYPE lolz
[
  <!ENTITY lol "lol">
  <!ELEMENT lolz (#PCDATA)>
  <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
  <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
  <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
  <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
  <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
  <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
  <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
  <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
  <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
  <!ENTITY lol10 "&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;&lol9;">
  <!ENTITY lol11 "&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10;&lol10
  <!ENTITY lol12 "&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11;&lol11
  <!ENTITY lol13 "&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12;&lol12
  <!ENTITY lol14 "&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13;&lol13
  <!ENTITY lol15 "&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14;&lol14
]>
<lolz>&lol15;</lolz>
```

A4:2017 - XML External Entities (XXE)





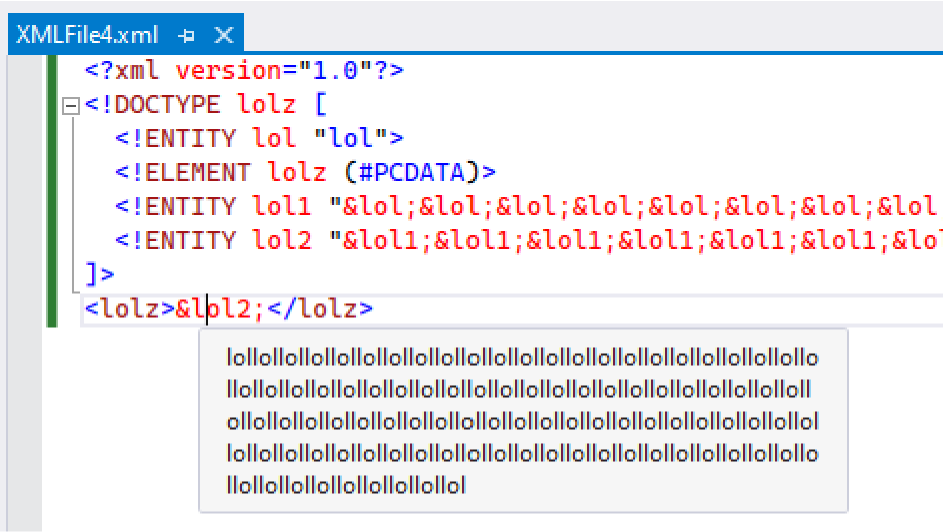
A4:2017 - XML External Entities (XXE)

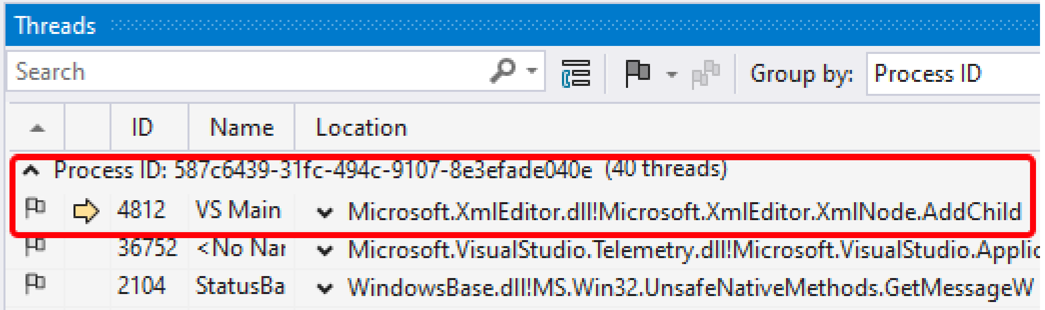
XMLFile4.xml

```
<?xml version="1.0"?>
<!DOCTYPE lolz [
  <!ENTITY lol "lol">
  <!ELEMENT lolz (#PCDATA)>
  <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol
  <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lo
]>
<lolz>&lol2;</lolz>
```

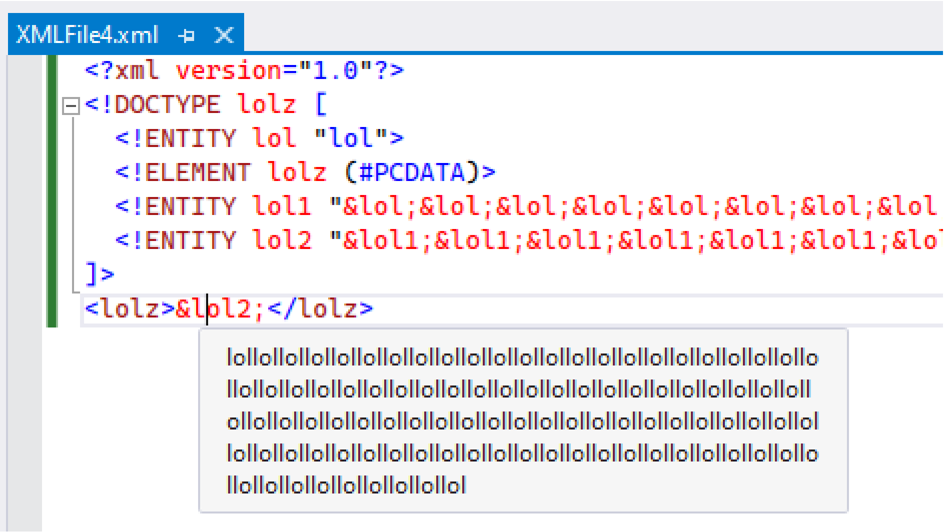
```
lollollollollollollollollollollollollollollollollollollollollollo
llollollollollollollollollollollollollollollollollollollollolloll
ollollollollollollollollollollollollollollollollollollollollollol
lollollollollollollollollollollollollollollollollollollollollollo
llollollollollollollol
```

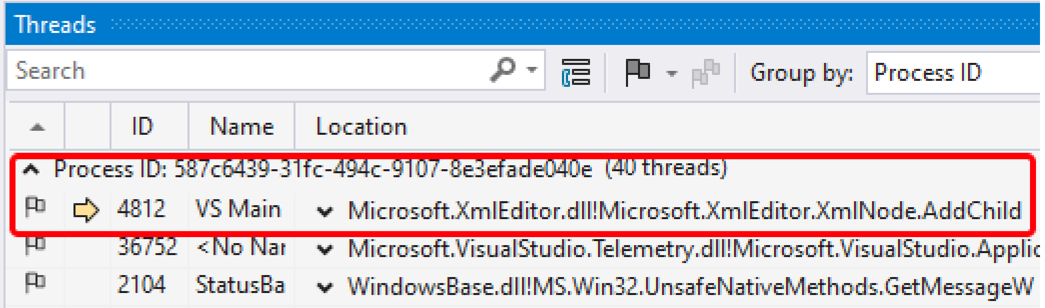
A4:2017 - XML External Entities (XXE)

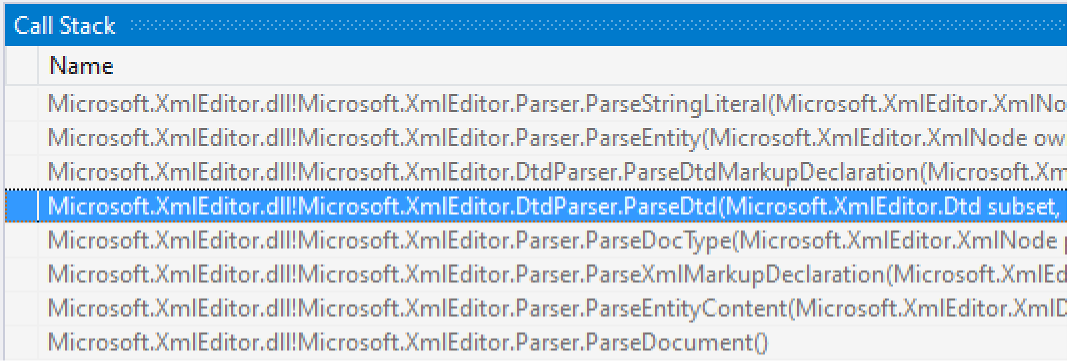




A4:2017 - XML External Entities (XXE)







Visual Studio 2022 и XML-бомбы

- CWE-776: Improper Restriction of Recursive Entity References in DTDs ('XML Entity Expansion’ (XEE))
- OWASP Top 10 2017 - A4:2017 - XML External Entities (XXE)
- OWASP Top 10 2021 - A05:2021 – Security Misconfiguration

Visual Studio 2022 и XML-бомбы

- CWE-776: Improper Restriction of Recursive Entity References in DTDs ('XML Entity Expansion’ (XEE))
- OWASP Top 10 2017 - A4:2017 - XML External Entities (XXE)
- OWASP Top 10 2021 - A05:2021 – Security Misconfiguration

- Исправлено после баг-репорта

AST (Application Security Testing)

- SAST
- DAST
- IAST
- SCA
- ....


PLEASE STAND BY

SAST

SAST: Static Application Security Testing

- Анализируем какое-то представление кода

SAST: Static Application Security Testing

- Анализируем какое-то представление кода
- Не требует исполнения приложения
  (его развёртывания – как следствие)

SAST: Static Application Security Testing

- Анализируем какое-то представление кода
- Не требует исполнения приложения
  (его развёртывания – как следствие)
- Покрывает всю кодовую базу
  (но не обязательно)

SAST: Static Application Security Testing

- Анализируем какое-то представление кода
- Не требует исполнения приложения
  (его развёртывания – как следствие)
- Покрывает всю кодовую базу
  (но не обязательно)
- Может давать (и даёт) false positive срабатывания

SAST: Static Application Security Testing

- Анализируем какое-то представление кода
- Не требует исполнения приложения
  (его развёртывания – как следствие)
- Покрывает всю кодовую базу
  (но не обязательно)
- Может давать (и даёт) false positive срабатывания
  (однако их количество можно сократить)

Syntax & semantic

А зачем?

А зачем?

- `a == a`

А зачем?

- `a == a`
- `(a) == a`

А зачем?

- `a == a`
- `(a) == a`
- `(a) == ((a))`

А зачем?

- `a == a`
- `(a) == a`
- `(a) == ((a))`
- `this.a == (a)`

А зачем?

- `a == a`
- `(a) == a`
- `(a) == ((a))`
- `this.a == (a)`
- `(base.a) == ((a))`

А зачем?

- `a == a`
- `(a) == a`
- `(a) == ((a))`
- `this.a == (a)`
- `(base.a) == ((a))`
- `(((((this.a))))) == ((base.a))`

А зачем?

- `a == a`
- `(a) == a`
- `(a) == ((a))`
- `this.a == (a)`
- `(base.a) == ((a))`
- `(((((this.a))))) == ((base.a))`
- `...`

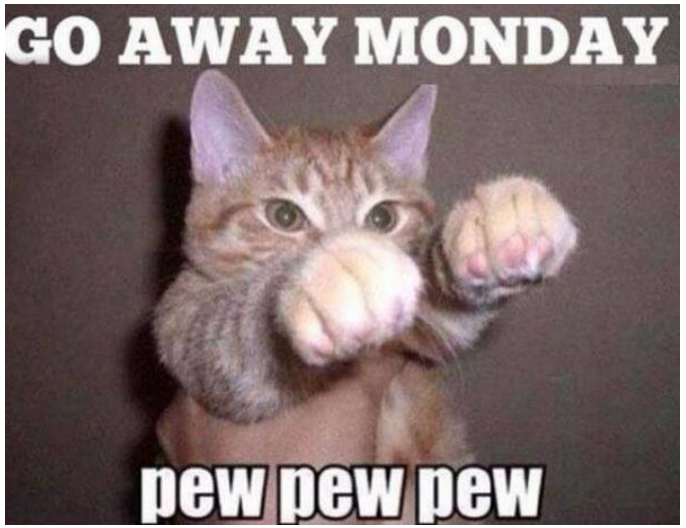
Syntax

Syntax

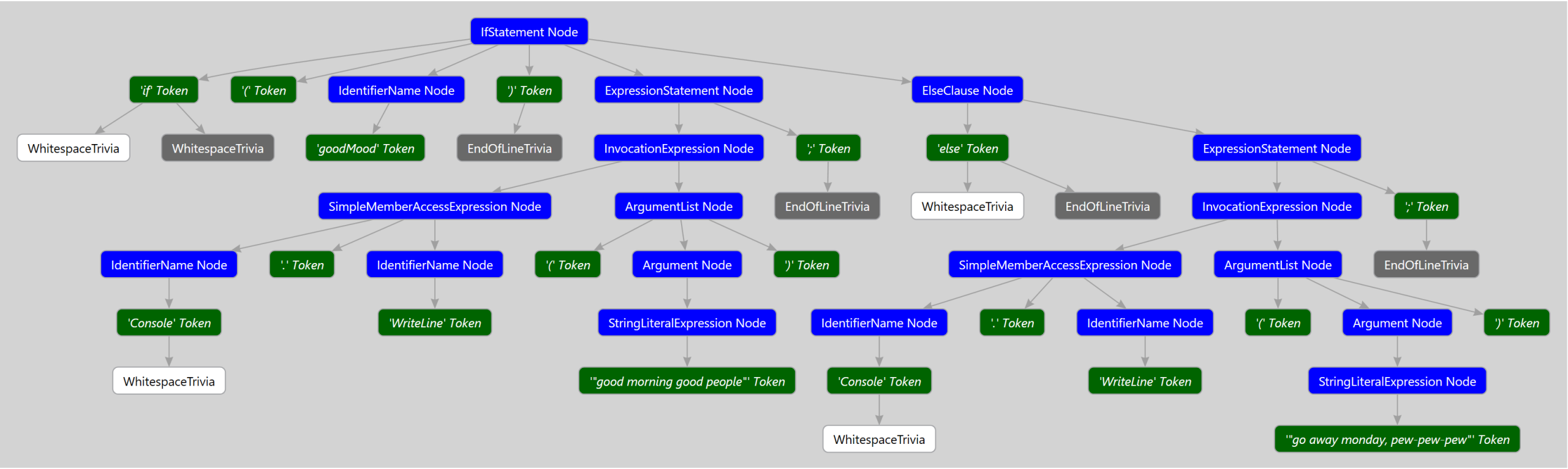
```
if (goodMood)
  Console.WriteLine("good morning good people");
else
  Console.WriteLine("go away Monday, pew-pew-pew");
```

Syntax

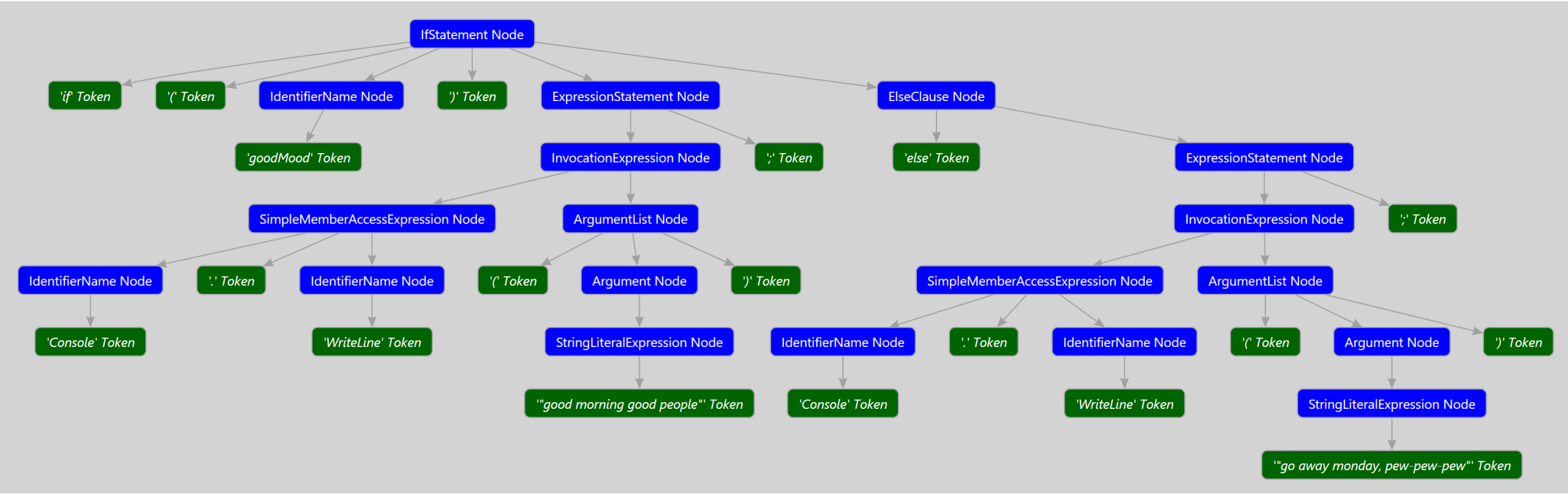
```
if (goodMood)
  Console.WriteLine("good morning good people");
else
  Console.WriteLine("go away Monday, pew-pew-pew");
```



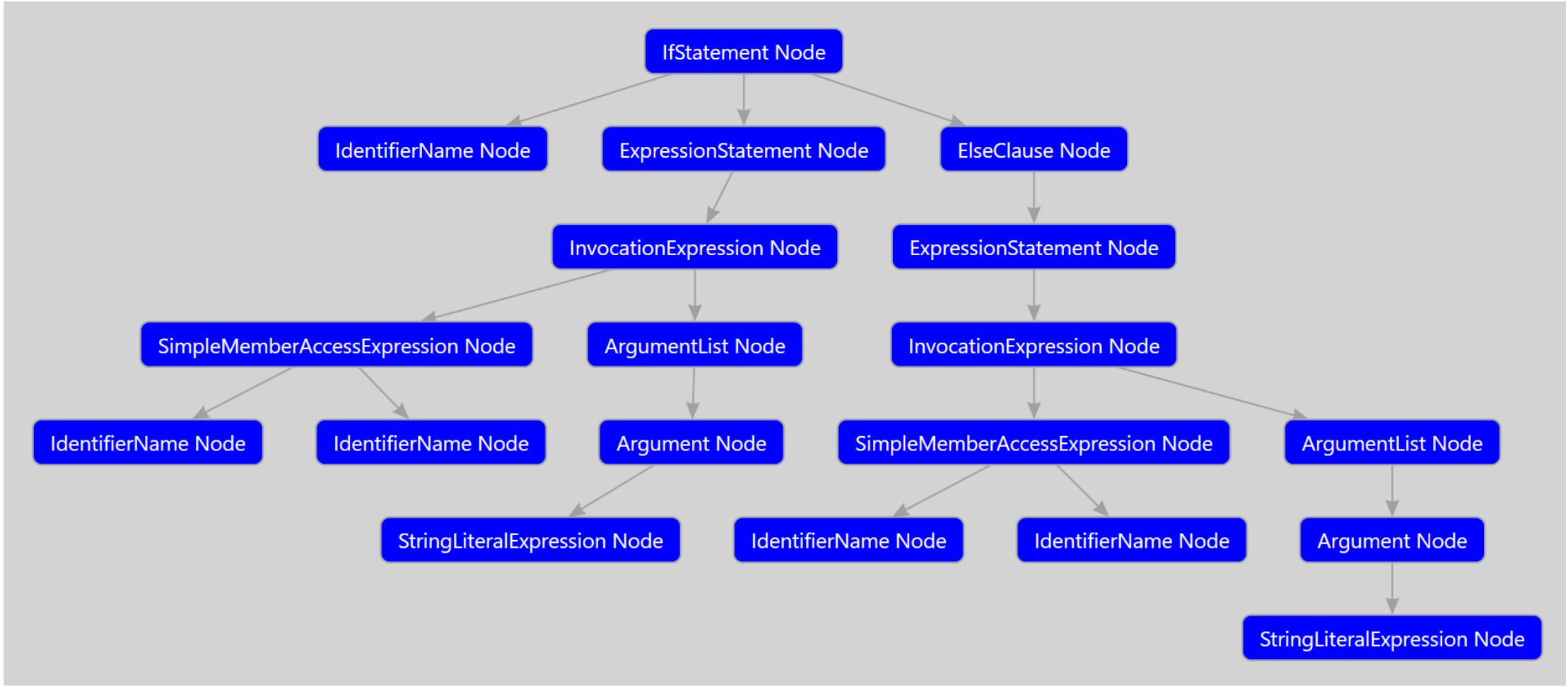
Syntax


IfStatement Node
'if' Token
'(' Token
IdentifierName Node
')' Token
ExpressionStatement Node
ElseClause Node
WhitespaceTrivia
WhitespaceTrivia
'goodMood' Token
EndOfLineTrivia
InvocationExpression Node
';' Token
'else' Token
ExpressionStatement Node
SimpleMemberAccessExpression Node
ArgumentList Node
EndOfLineTrivia
WhitespaceTrivia
EndOfLineTrivia
InvocationExpression Node
';' Token
IdentifierName Node
'.' Token
IdentifierName Node
'(' Token
Argument Node
')' Token
SimpleMemberAccessExpression Node
ArgumentList Node
EndOfLineTrivia
'Console' Token
'WriteLine' Token
StringLiteralExpression Node
IdentifierName Node
'.' Token
IdentifierName Node
'(' Token
Argument Node
')' Token
WhitespaceTrivia
'"good morning good people"' Token
'Console' Token
'WriteLine' Token
StringLiteralExpression Node
WhitespaceTrivia
'"go away monday, pew-pew-pew"' Token

Syntax


IfStatement Node
'if' Token
'(' Token
IdentifierName Node
')' Token
ExpressionStatement Node
ElseClause Node
'goodMood' Token
InvocationExpression Node
';' Token
'else' Token
ExpressionStatement Node
SimpleMemberAccessExpression Node
ArgumentList Node
InvocationExpression Node
';' Token
IdentifierName Node
'.' Token
IdentifierName Node
'(' Token
Argument Node
')' Token
SimpleMemberAccessExpression Node
ArgumentList Node
'Console' Token
'WriteLine' Token
StringLiteralExpression Node
IdentifierName Node
'.' Token
IdentifierName Node
'(' Token
Argument Node
')' Token
'"good morning good people"' Token
'Console' Token
'WriteLine' Token
StringLiteralExpression Node
'"go away monday, pew-pew-pew"' Token

Syntax


IfStatement Node
IdentifierName Node
ExpressionStatement Node
ElseClause Node
InvocationExpression Node
ExpressionStatement Node
SimpleMemberAccessExpression Node
ArgumentList Node
InvocationExpression Node
IdentifierName Node
IdentifierName Node
Argument Node
SimpleMemberAccessExpression Node
ArgumentList Node
StringLiteralExpression Node
IdentifierName Node
IdentifierName Node
Argument Node
StringLiteralExpression Node

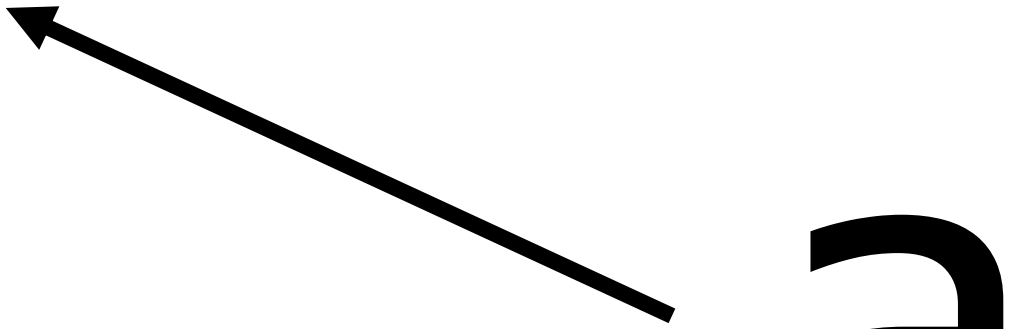
Semantic

Semantic

a

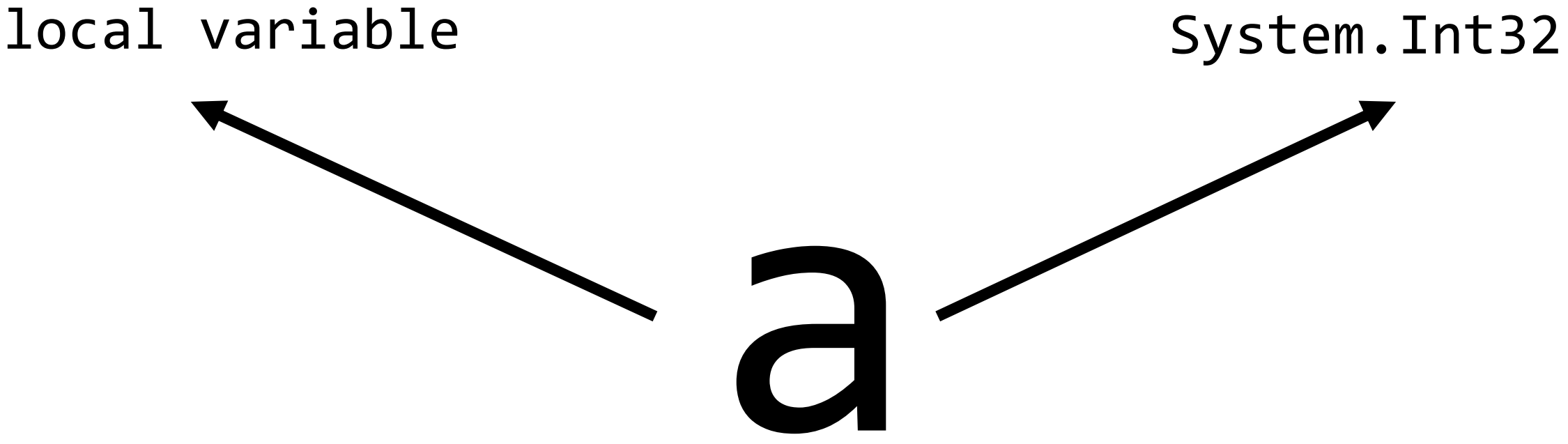
Semantic

```
local variable
```



a

Semantic



Syntax & semantic

- Средства для более удобной работы с кодом (особенно если API для людей. Roslyn - <3)

Syntax & semantic

- Средства для более удобной работы с кодом (особенно если API для людей. Roslyn - <3)
- Семантика: информация о сущностях, типах и т.п.

Syntax & semantic

- Средства для более удобной работы с кодом (особенно если API для людей. Roslyn - <3)
- Семантика: информация о сущностях, типах и т.п.
- Достаточно для отлова многих багов

iOS (CVE-2014-1266)

```
if ((err = SSLHashSHA1.update(
                &hashCtx, &signedParams)) != 0)
  goto fail;
  goto fail;
```

iOS (CVE-2014-1266)

```
if ((err = SSLHashSHA1.update(
                &hashCtx, &signedParams)) != 0)
    goto fail;
    goto fail;
```

iOS (CVE-2014-1266)

```
if ((err = SSLHashSHA1.update(
                &hashCtx, &signedParams)) != 0)
  goto fail;
goto fail;
```

iOS (CVE-2014-1266)

```
if ((err = SSLHashSHA1.update(
                &hashCtx, &signedParams)) != 0)
  goto fail;
  goto fail;
```

MySQL (CVE-2012-2122)

```
typedef char my_bool;
my_bool
check_scramble(const char *scramble_arg,
               const char *message,
               const uint8 *hash_stage2) {

  ....
  return memcmp(hash_stage2,
                hash_stage2_reassured,
                SHA1_HASH_SIZE);
}
```

int -> char

Data-flow analysis

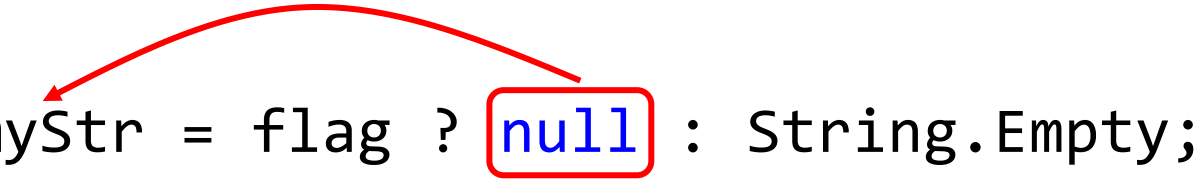
Data-flow analysis

```
var myStr = flag ? null : String.Empty;
....
if (anotherFlag)
{
   ....
   var len = myStr.Length;
}
```

Data-flow analysis

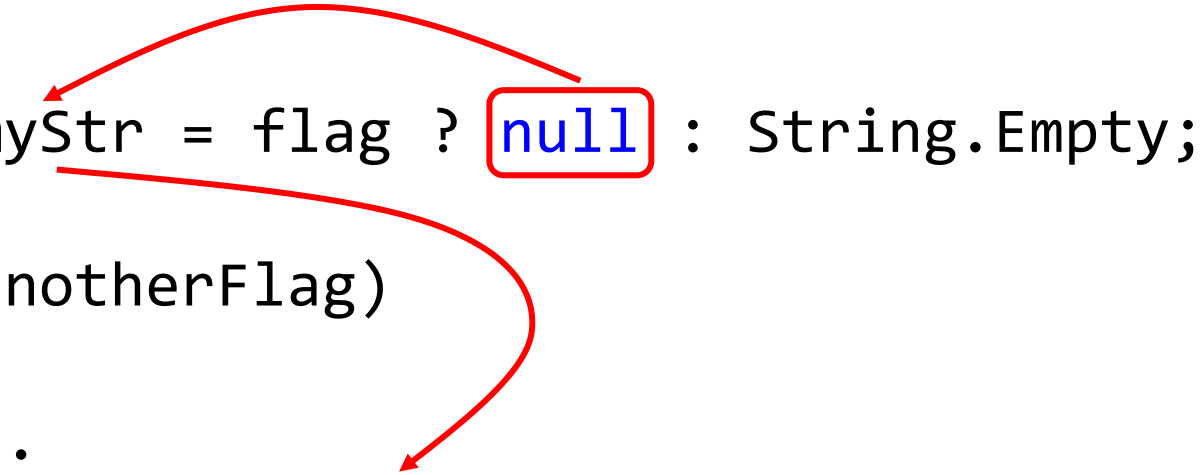
```
var myStr = flag ? null : String.Empty;
....
if (anotherFlag)
{
   ....
   var len = myStr.Length;
}
```

Data-flow analysis



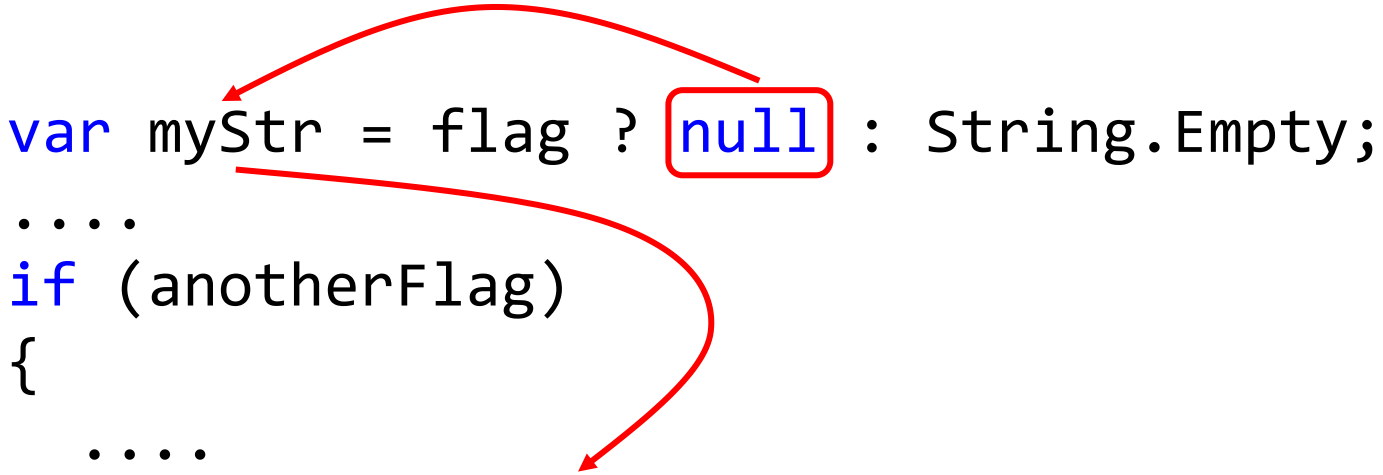
```
var myStr = flag ? null : String.Empty;
....
if (anotherFlag)
{
   ....
   var len = myStr.Length;
}
```

Data-flow analysis



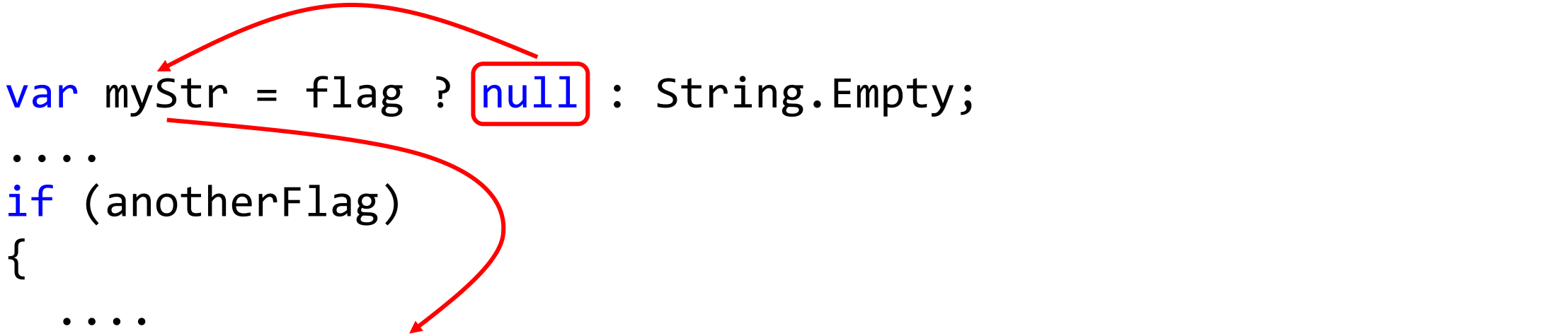
```
var myStr = flag ? null : String.Empty;
....
if (anotherFlag)
{
   ....
   var len = myStr.Length;
}
```

Data-flow analysis



```
var myStr = flag ? null : String.Empty;
....
if (anotherFlag)
{
   ....
   var len = myStr.Length;
}
```

Data-flow analysis



```
var myStr = flag ? null : String.Empty;
....
if (anotherFlag)
{
   ....
   var len = myStr.Length;  // Possible null reference exception
}
```

Data-flow analysis

```
int mappingClassCount = this.Mapping.GetClassCountSafe(....);
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

Data-flow analysis

```
int mappingClassCount = this.Mapping.GetClassCountSafe(....);
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = mapping.GetClassCount(instanceSource,
                                         labelSource);
  if (classCount < 2) {
    throw new MappingException(....);
  }

  return classCount;
}
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = ....;

  if (classCount < 2) {
    throw ....;
  }

  return classCount;
}
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = ....;

  if (classCount < 2) {
    throw ....;
  }

  return classCount;
}
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = ....; [int.MinValue .. int.MaxValue]

  if (classCount < 2) {
    throw ....;
  }

  return classCount;
}
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = ....; [int.MinValue .. int.MaxValue]

  if (classCount < 2) {
    throw ....;
  }

  return classCount;
}
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = ....; [int.MinValue .. int.MaxValue]

  if (classCount < 2) {
    throw ....;              [int.MinValue .. 1]
  }

  return classCount;
}
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = ....; [int.MinValue .. int.MaxValue]

  if (classCount < 2) {
    throw ....;              [int.MinValue .. 1]
  }

  return classCount;
}
```

Data-flow analysis

```
public static int GetClassCountSafe<....>(....)
{
  int classCount = ....; [int.MinValue .. int.MaxValue]

  if (classCount < 2) {
    throw ....;              [int.MinValue .. 1]
  }

  return classCount;         [2 .. int.MaxValue]
}
```

Data-flow analysis

```
int mappingClassCount = this.Mapping.GetClassCountSafe(....);
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

Data-flow analysis

```
int mappingClassCount = ....; [2 .. int.MaxValue]
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

Data-flow analysis

```
int mappingClassCount = ....; [2 .. int.MaxValue]
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

Data-flow analysis

```
int mappingClassCount = ....; [2 .. int.MaxValue]
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

Data-flow analysis

```
int mappingClassCount = ....; [2 .. int.MaxValue]
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

Data-flow analysis

```
int mappingClassCount = ....; [2 .. int.MaxValue]
....
if (mappingClassCount == 2) {
  throw new BayesPointMachineClassifierException(....);
}
if (mappingClassCount < 2) {
  throw new BayesPointMachineClassifierException(....);
}
....
```

CWE-570: Expression is Always False

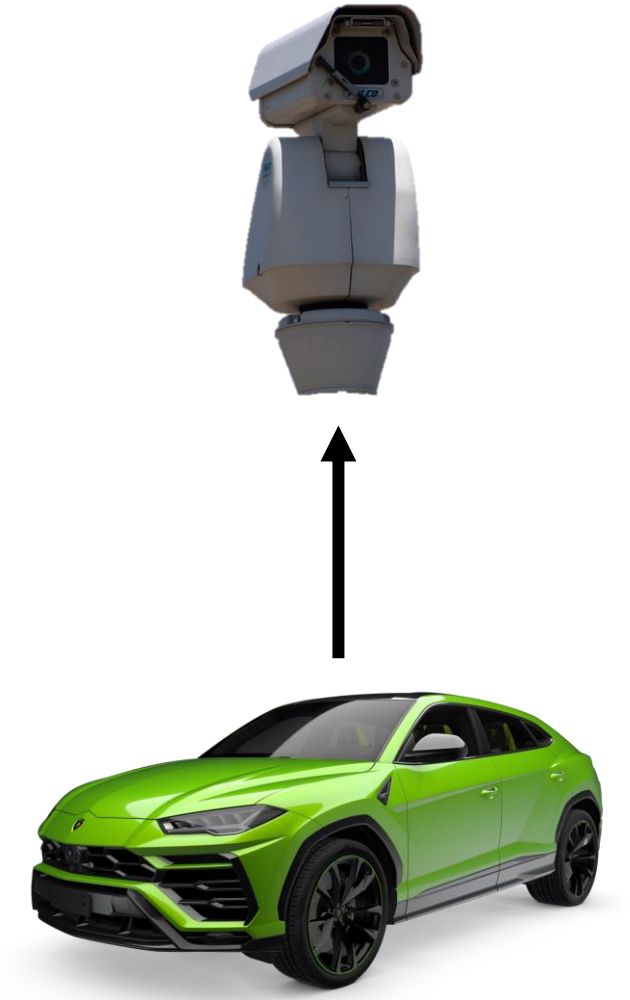
SQLI (SQL injection)

SQL injection

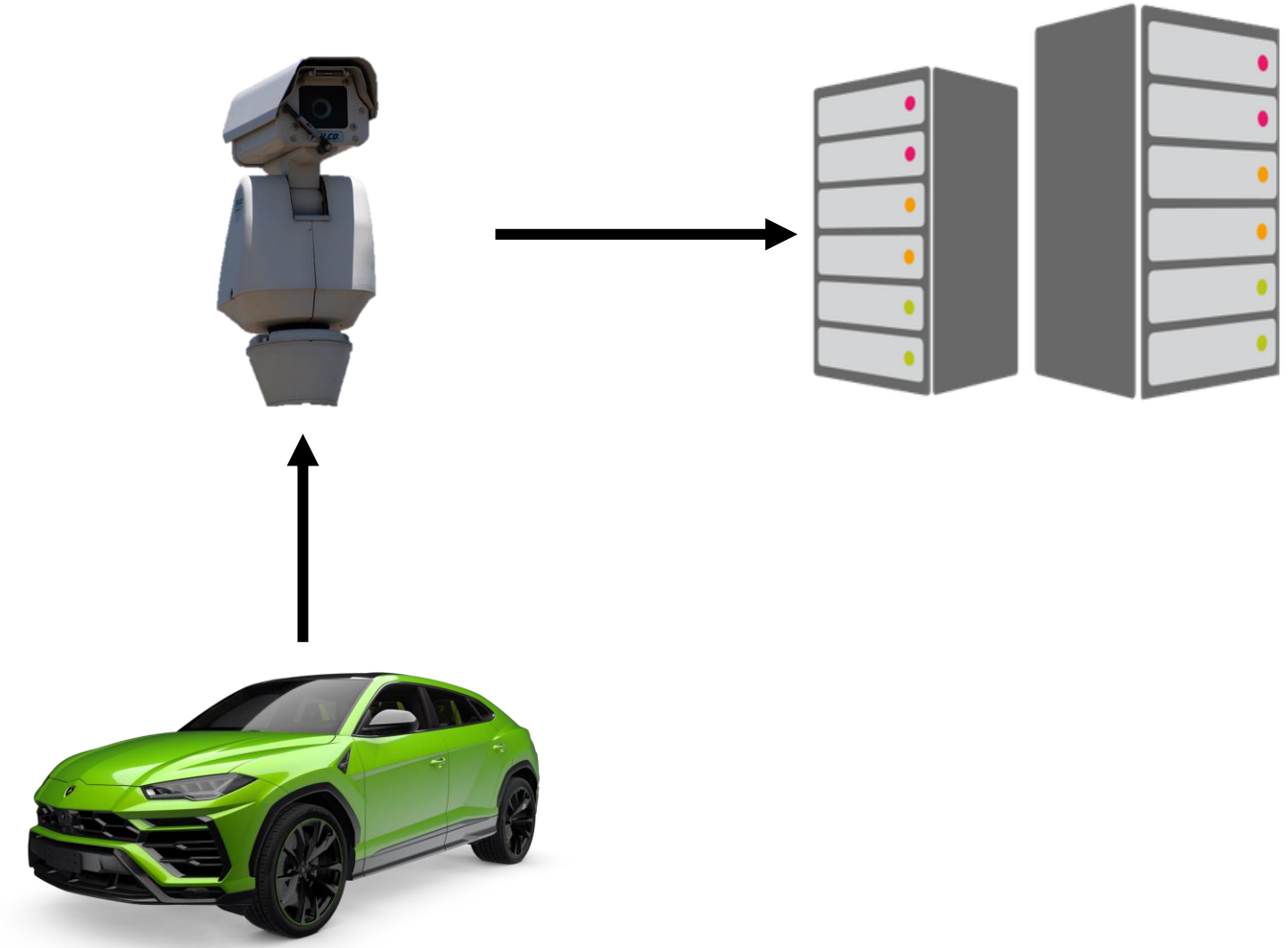
SQL injection

SQL injection

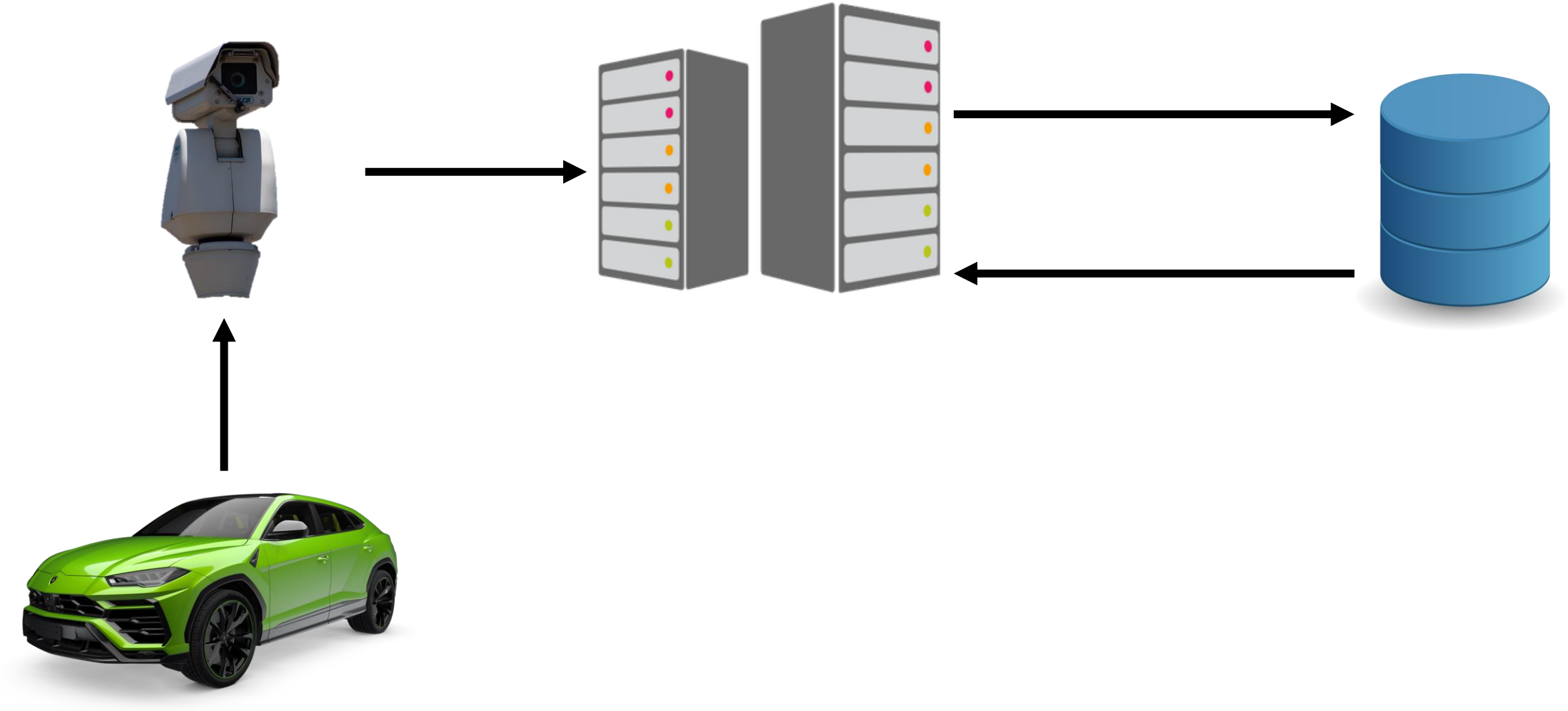
SQL injection



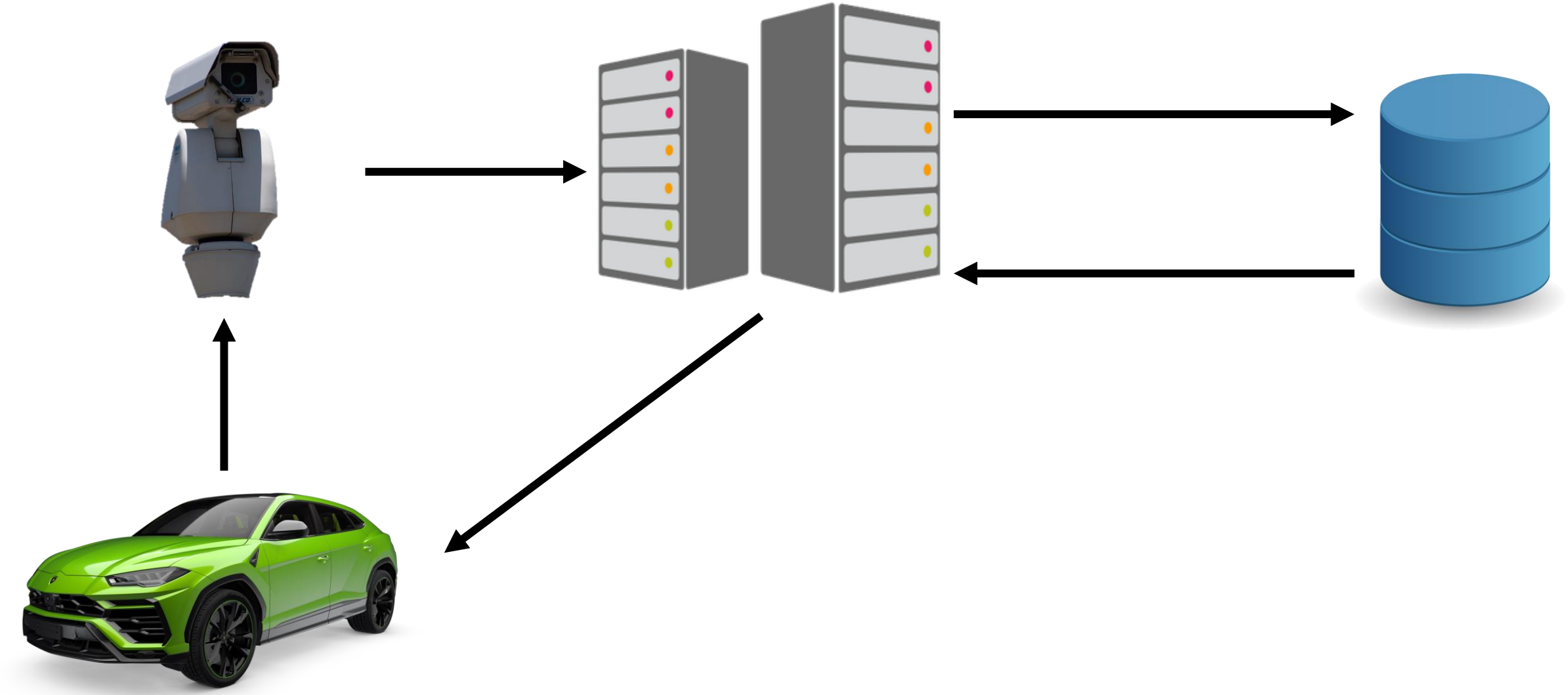
SQL injection



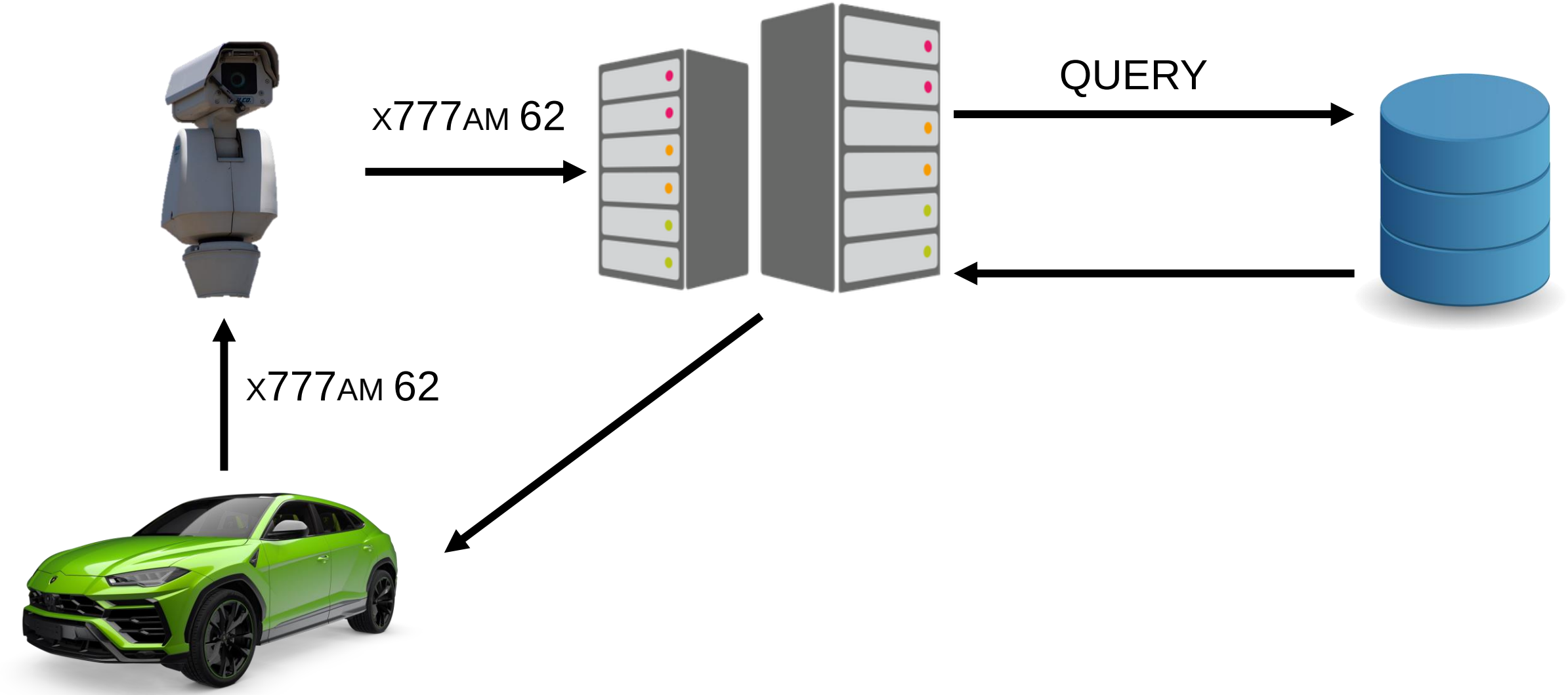
SQL injection



SQL injection



SQL injection



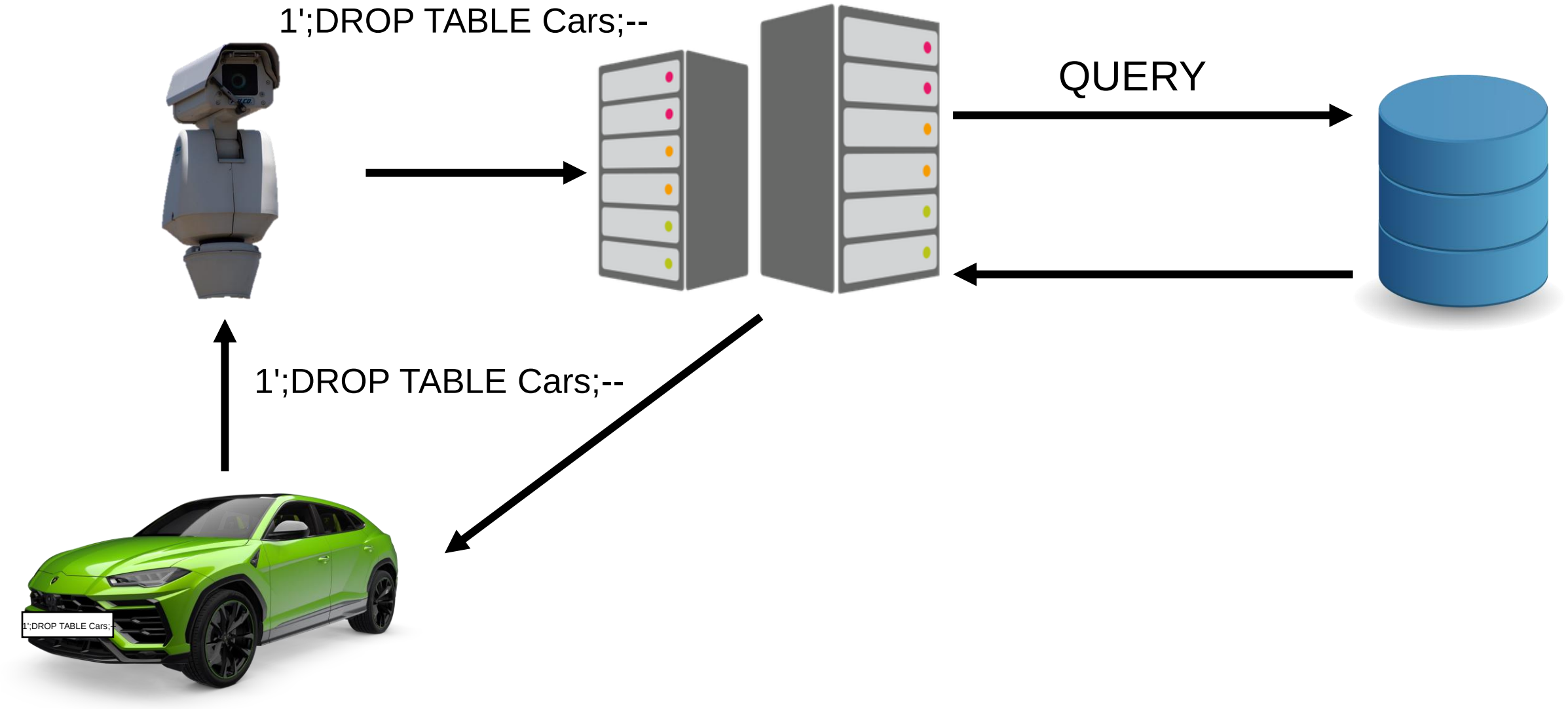
SQL injection

```
SELECT * FROM Cars WHERE PlateNumber = ' + PlateNumber + '
```

SQL injection

```
SELECT * FROM Cars WHERE PlateNumber = ' + PlateNumber + '

// x777am62
SELECT * FROM Cars WHERE PlateNumber = 'x777am62'
```

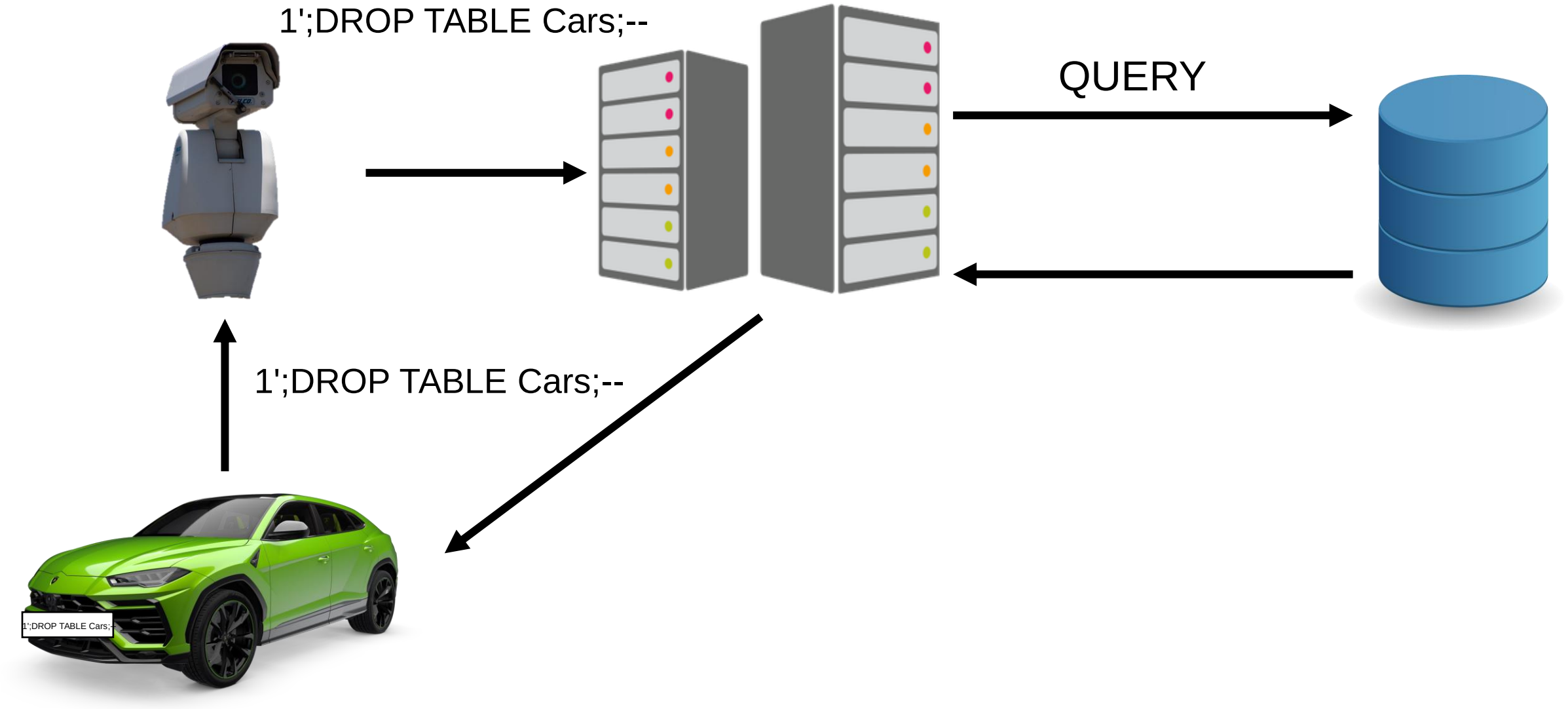
SQL injection



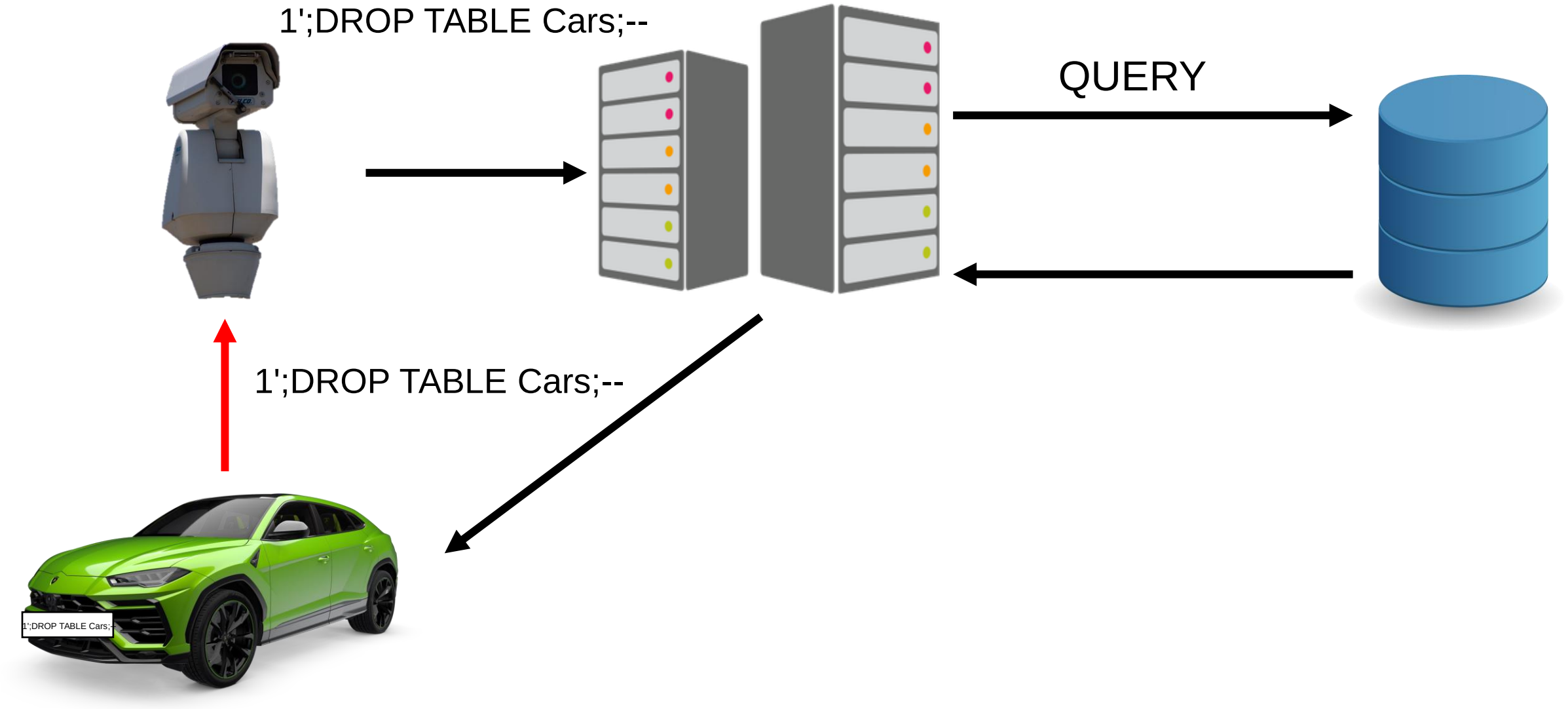
SQL injection

```
SELECT * FROM Cars WHERE PlateNumber = ' + PlateNumber + '

// x777am62
SELECT * FROM Cars WHERE PlateNumber = 'x777am62'

// 1';DROP TABLE Cars;--
SELECT * FROM Cars WHERE PlateNumber = '1';DROP TABLE Cars;--
```

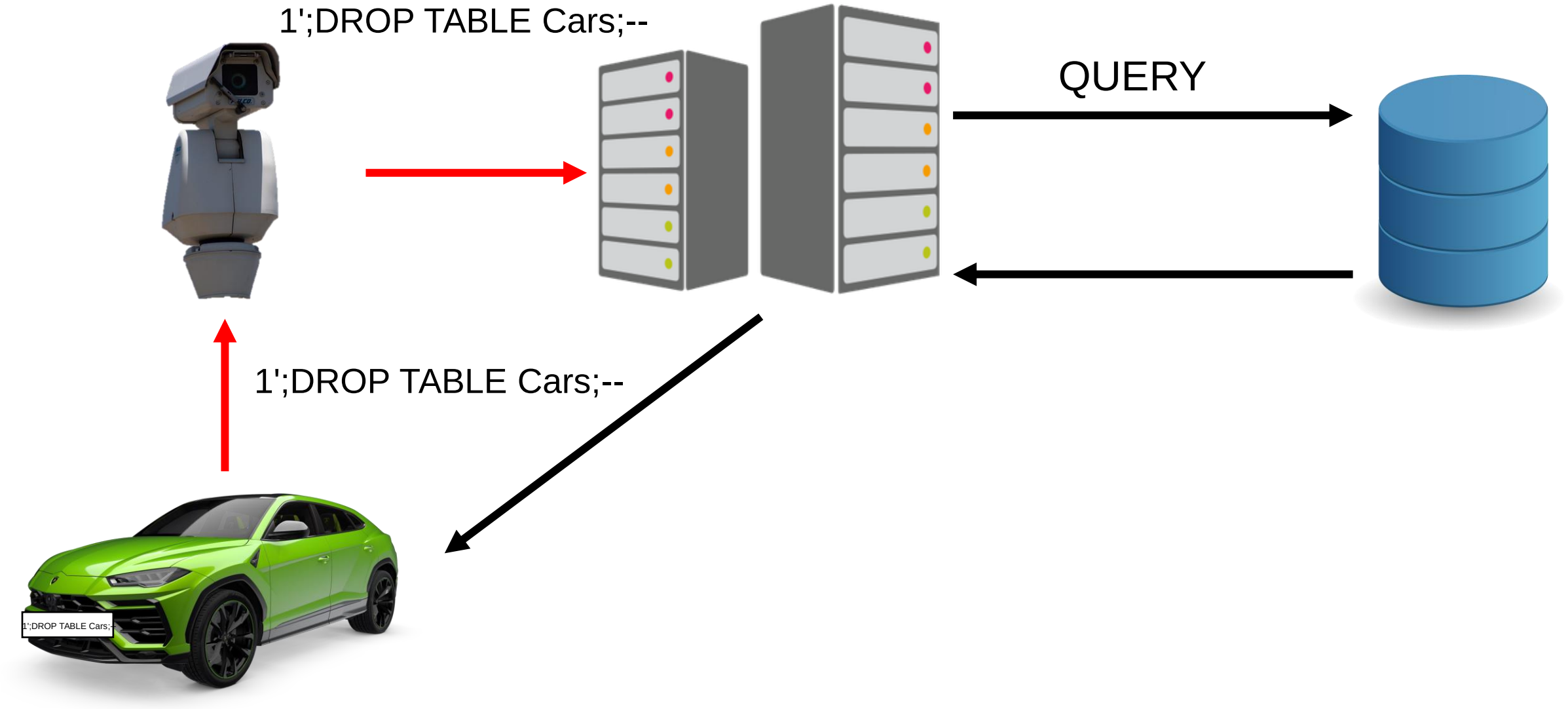
SQL injection



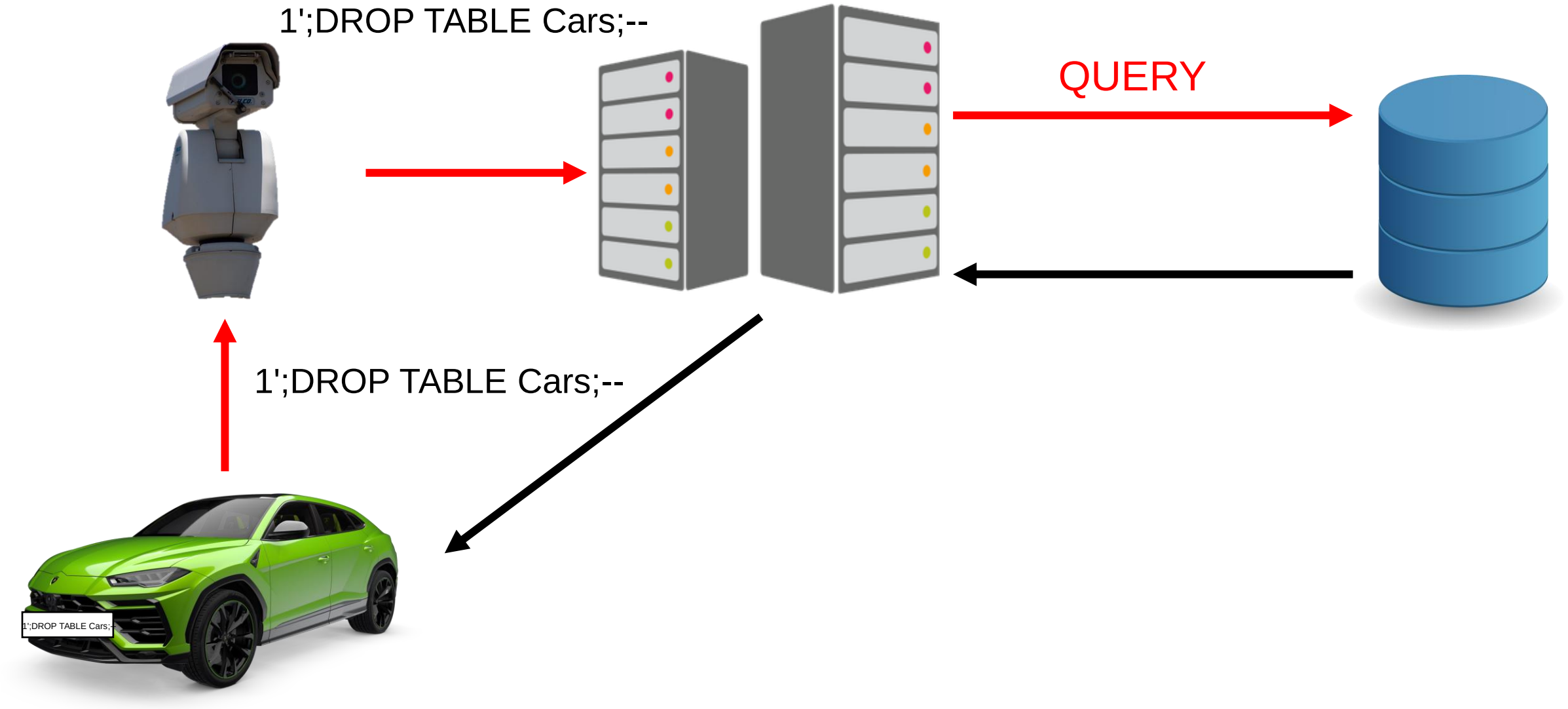
SQL injection



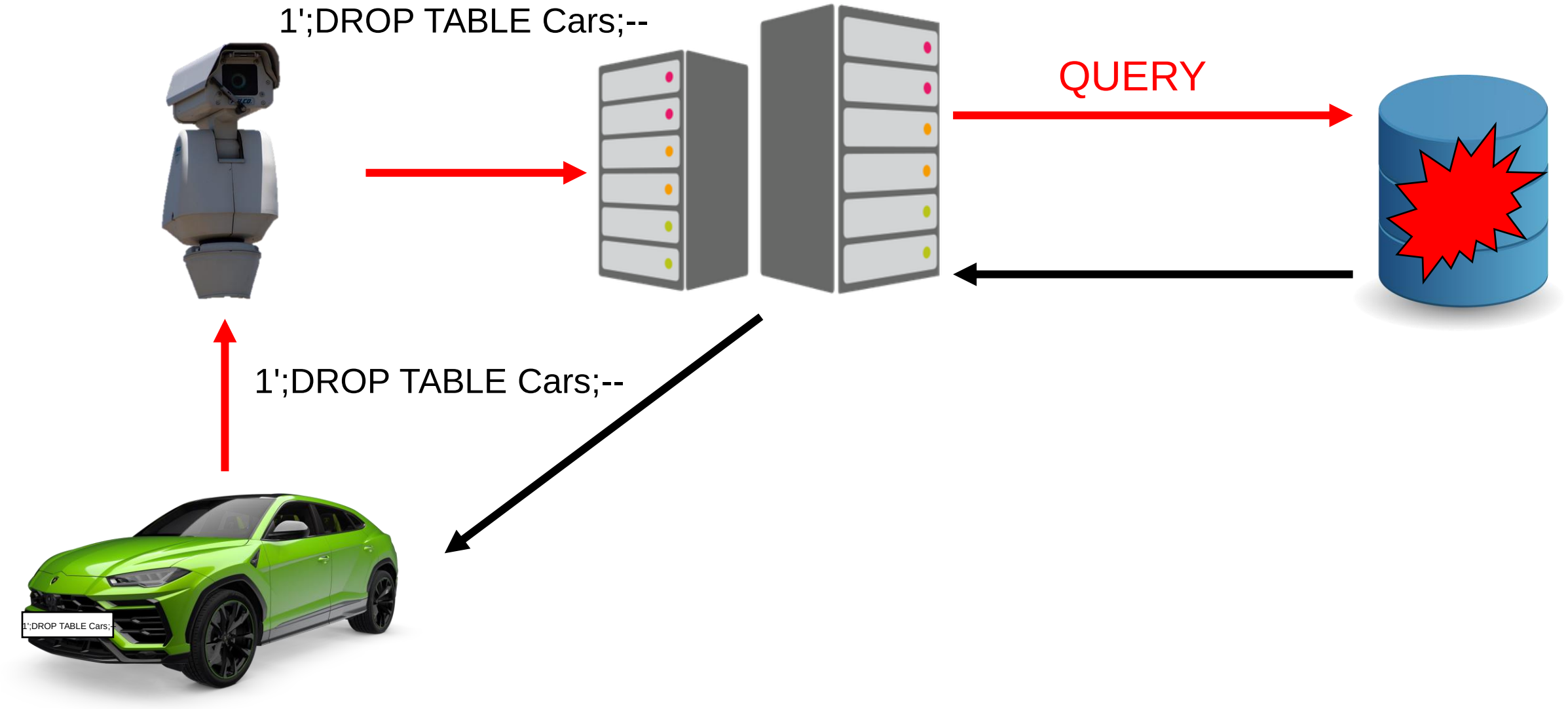
SQL injection



SQL injection



SQL injection



SQLI

```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

SQLI

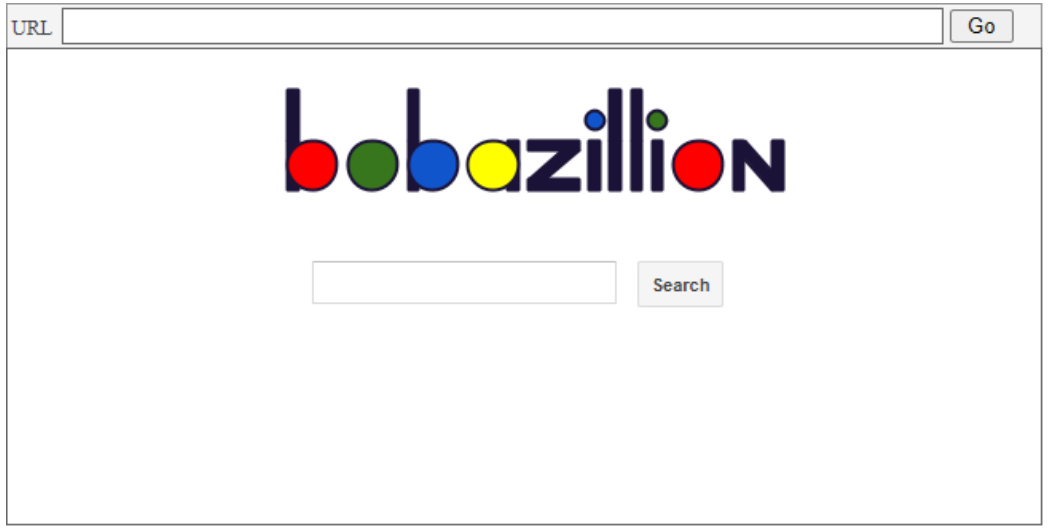
```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

SQLI

```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

XSS (cross-site scripting)

XSS (cross-site scripting)



XSS (cross-site scripting)

Input: Why Skopin is the best city?

XSS (cross-site scripting)

Input: Why Skopin is the best city?

XSS (cross-site scripting)

Input: <s>Skopin</s>

XSS (cross-site scripting)

Input: <s>Skopin</s>

XSS (cross-site scripting)

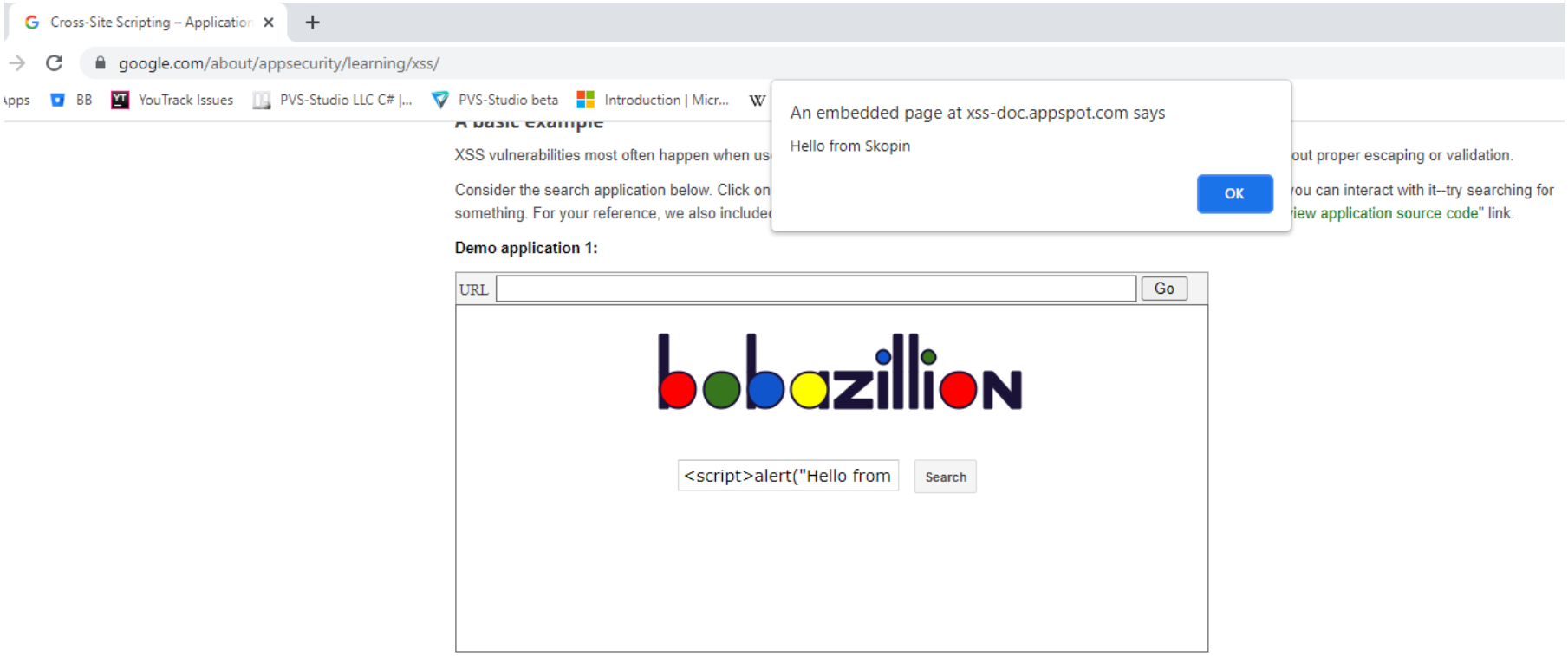
Input: <s>Skopin</s>

XSS (cross-site scripting)

Input: <script>alert("Hello from Skopin")</script >

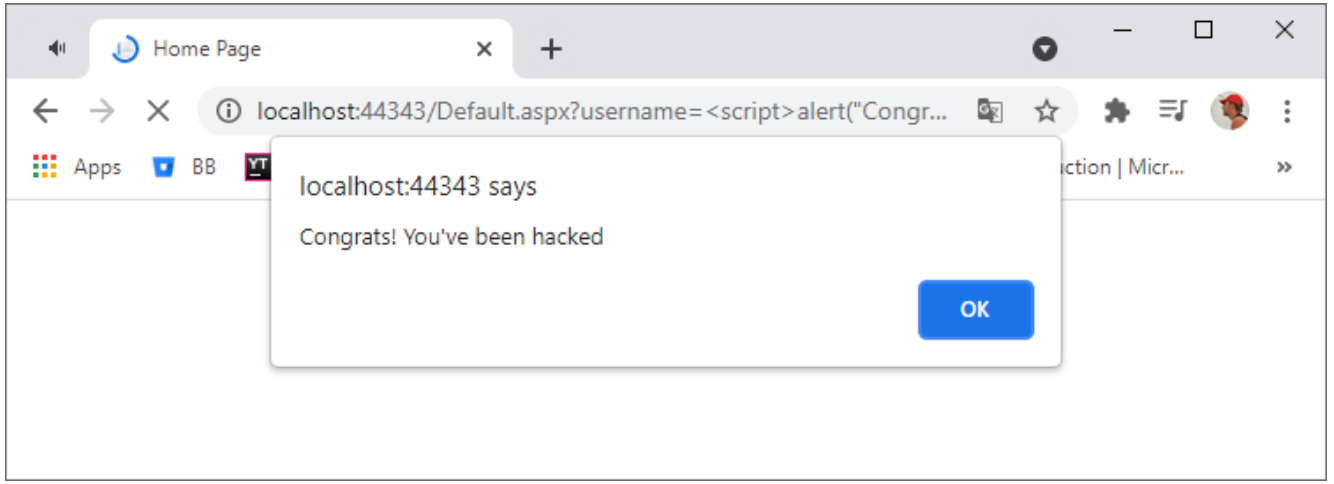
XSS (cross-site scripting)

Input: <script>alert("Hello from Skopin")</script >



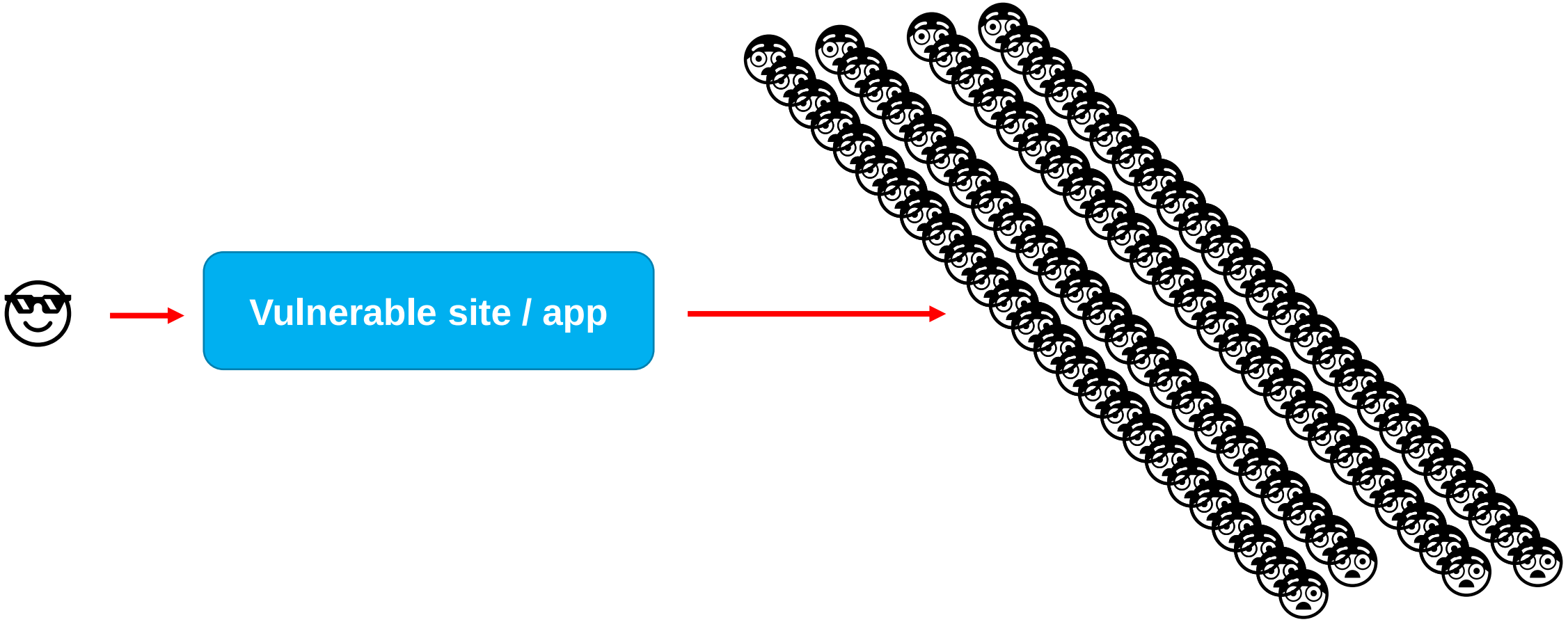
XSS (cross-site scripting)

Input: ...?username=<script>alert("Congrats! You've been hacked")</script>



XSS (cross-site scripting)

XSS (cross-site scripting)



XSS

```
protected void Page_Load(object sender, EventArgs e)
{
  var userName = Request.Params["userName"];

  string message;
  if (string.IsNullOrWhiteSpace(userName))
  {
      message = string.Format(_centerAlignFormat,
                              "Empty 'userName' parameter");
  }
  else
  {
      message = string.Format(_centerAlignFormat,
                              $"'{userName}' data has been processed.");
  }

  Response.Write(message);
}
```

XSS

```
protected void Page_Load(object sender, EventArgs e)
{
  var userName = Request.Params["userName"];

  string message;
  if (string.IsNullOrWhiteSpace(userName))
  {
      message = string.Format(_centerAlignFormat,
                              "Empty 'userName' parameter");
  }
  else
  {
      message = string.Format(_centerAlignFormat,
                              $"'{userName}' data has been processed.");
  }

  Response.Write(message);
}
```

XSS

```
protected void Page_Load(object sender, EventArgs e)
{
  var userName = Request.Params["userName"];

  string message;
  if (string.IsNullOrWhiteSpace(userName))
  {
      message = string.Format(_centerAlignFormat,
                              "Empty 'userName' parameter");
  }
  else
  {
      message = string.Format(_centerAlignFormat,
                              $"'{userName}' data has been processed.");
  }

  Response.Write(message);
}
```

XSS

```
protected void Page_Load(object sender, EventArgs e)
{
  var userName = Request.Params["userName"];

  string message;
  if (string.IsNullOrWhiteSpace(userName))
  {
      message = string.Format(_centerAlignFormat,
                              "Empty 'userName' parameter");
  }
  else
  {
      message = string.Format(_centerAlignFormat,
                              $"'{userName}' data has been processed.");
  }

  Response.Write(message);
}
```

Излишнее доверие к внешним данным

Проектирование по оптимистичному сценарию

Хакер, когда ты проектируешь по оптимистичному сценарию


PLEASE STAND BY

Taint analysis (taint checking)

Taint analysis

- Проблема излишнего доверия к входным данным

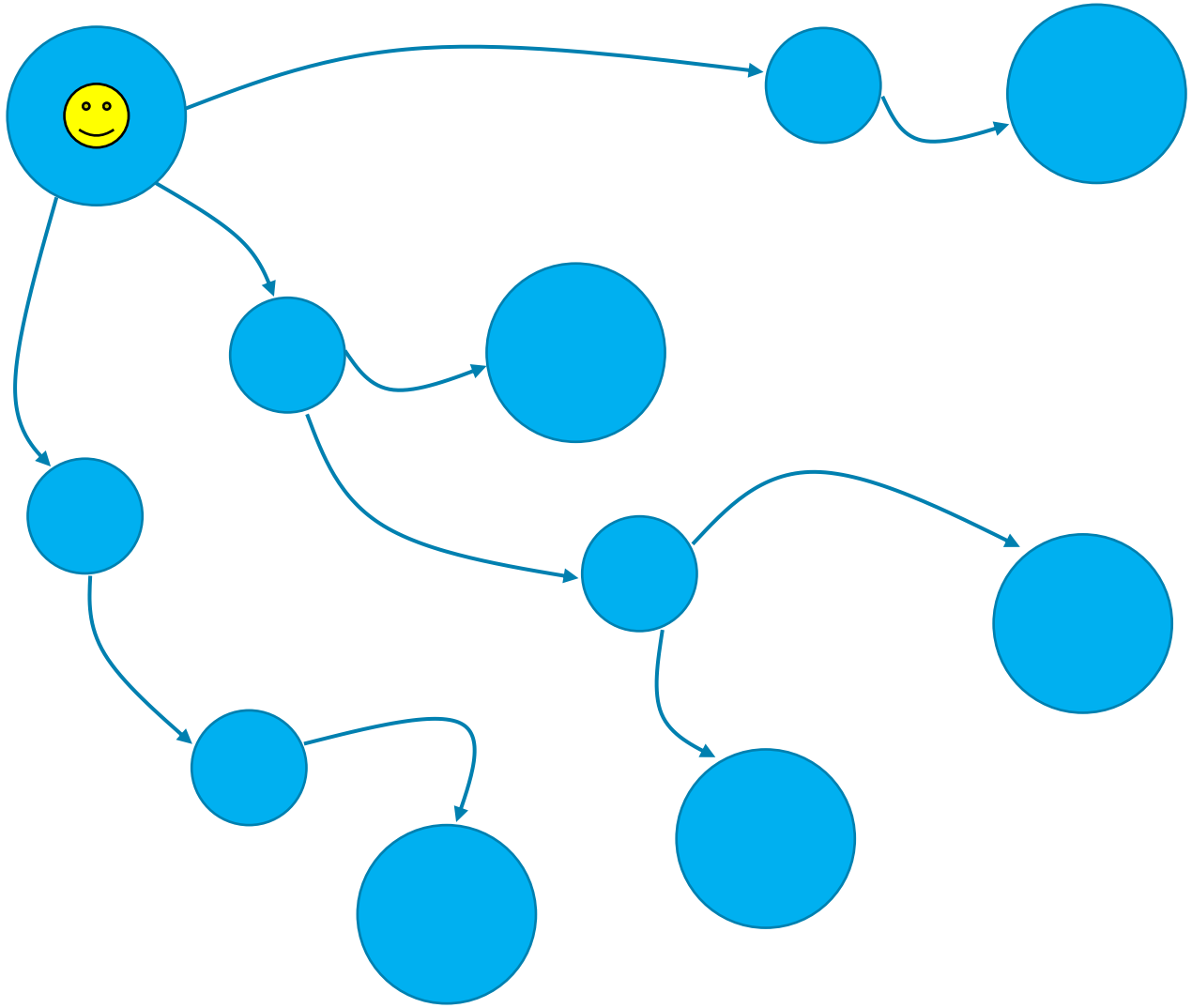
Taint analysis

- Проблема излишнего доверия к входным данным
- Помогает в поиске:
  - SQL injection
  - OS command injection
  - XSS (cross-site scripting)
  - path traversal
  - XXE и XEE
  - и т.п.

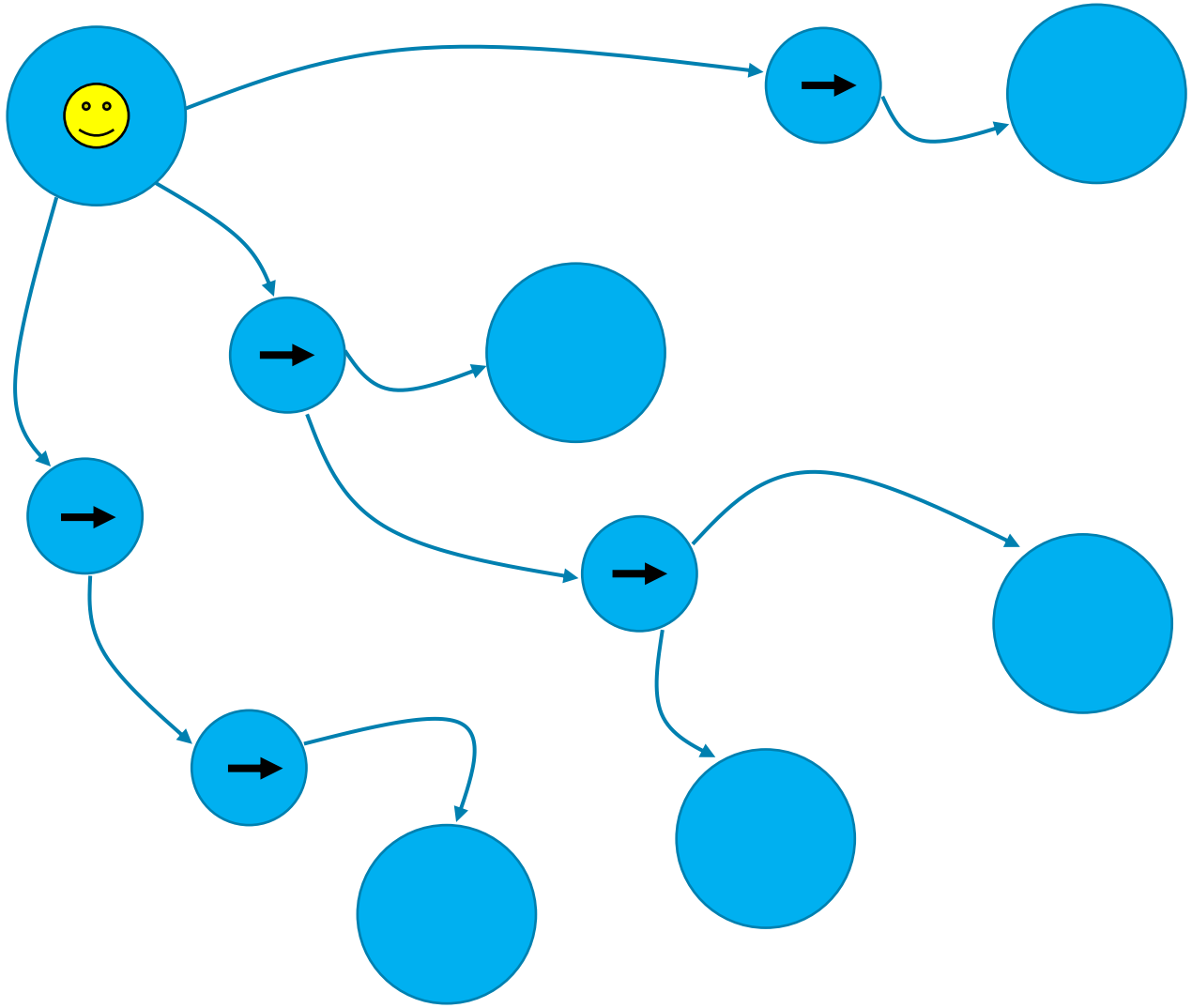
Taint analysis

- Проблема излишнего доверия к входным данным
- Помогает в поиске:
  - SQL injection
  - OS command injection
  - XSS (cross-site scripting)
  - path traversal
  - XXE и XEE
  - и т.п.
- CWE, OWASP, Top'ы...

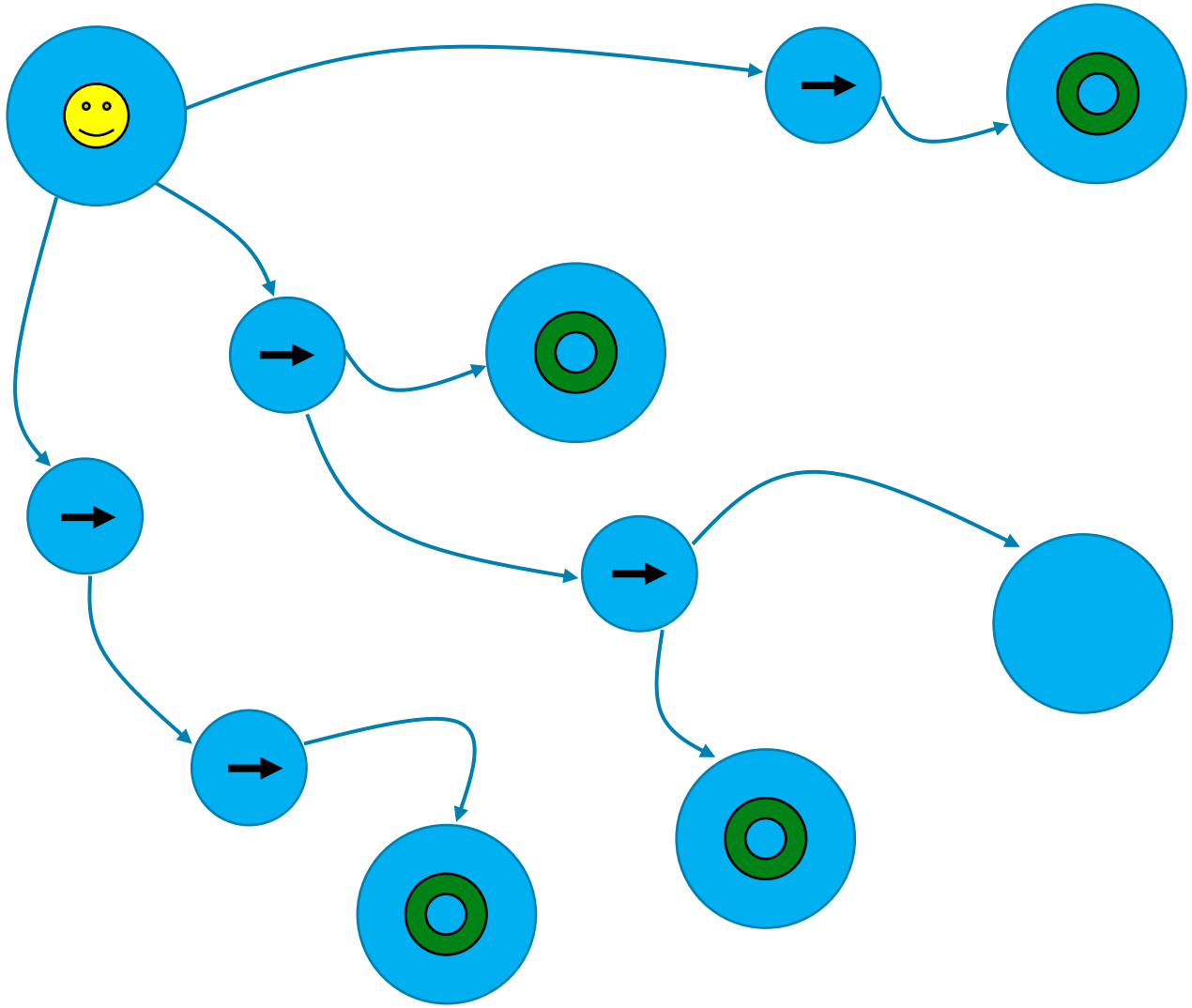
Taint analysis



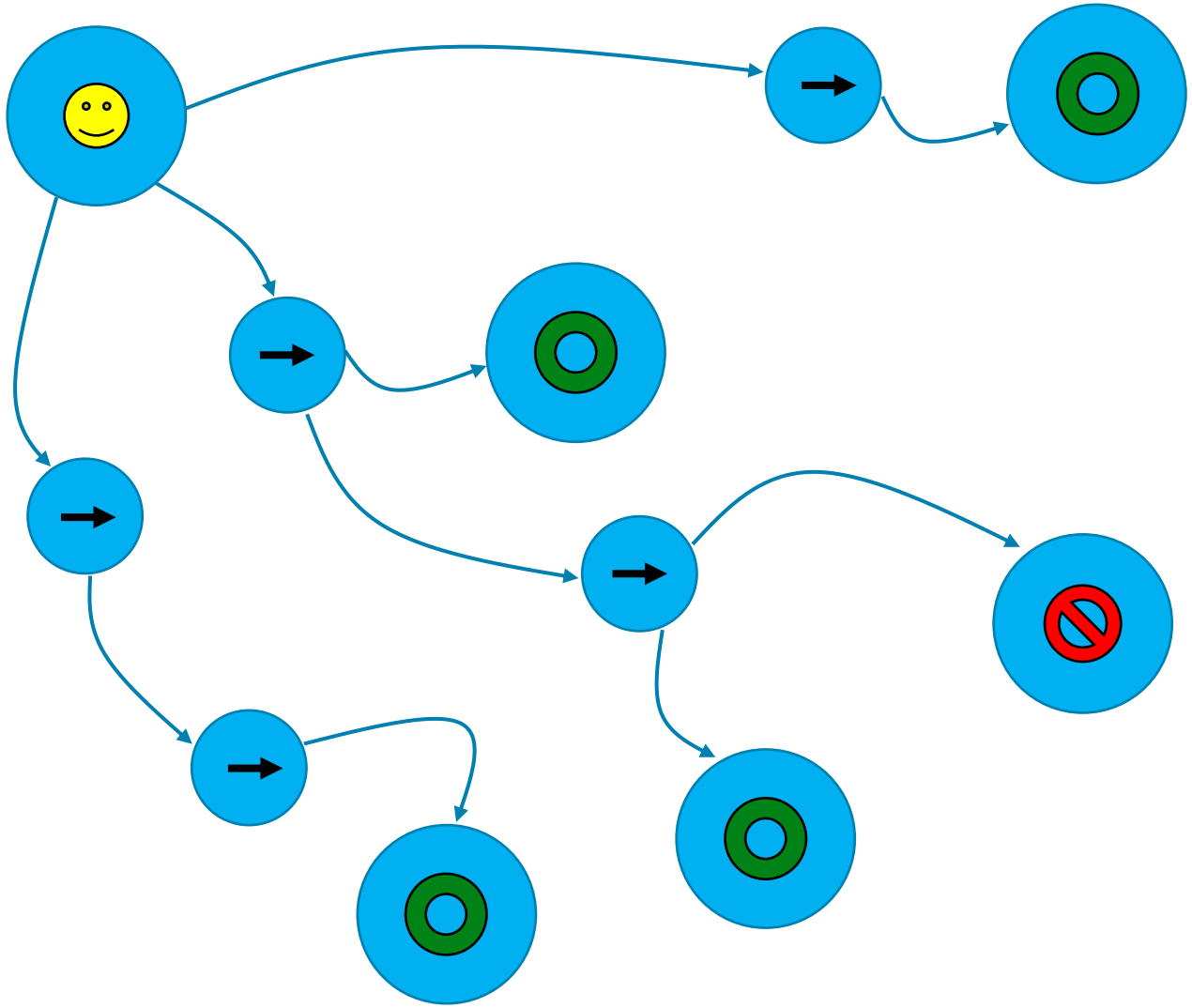
Taint analysis



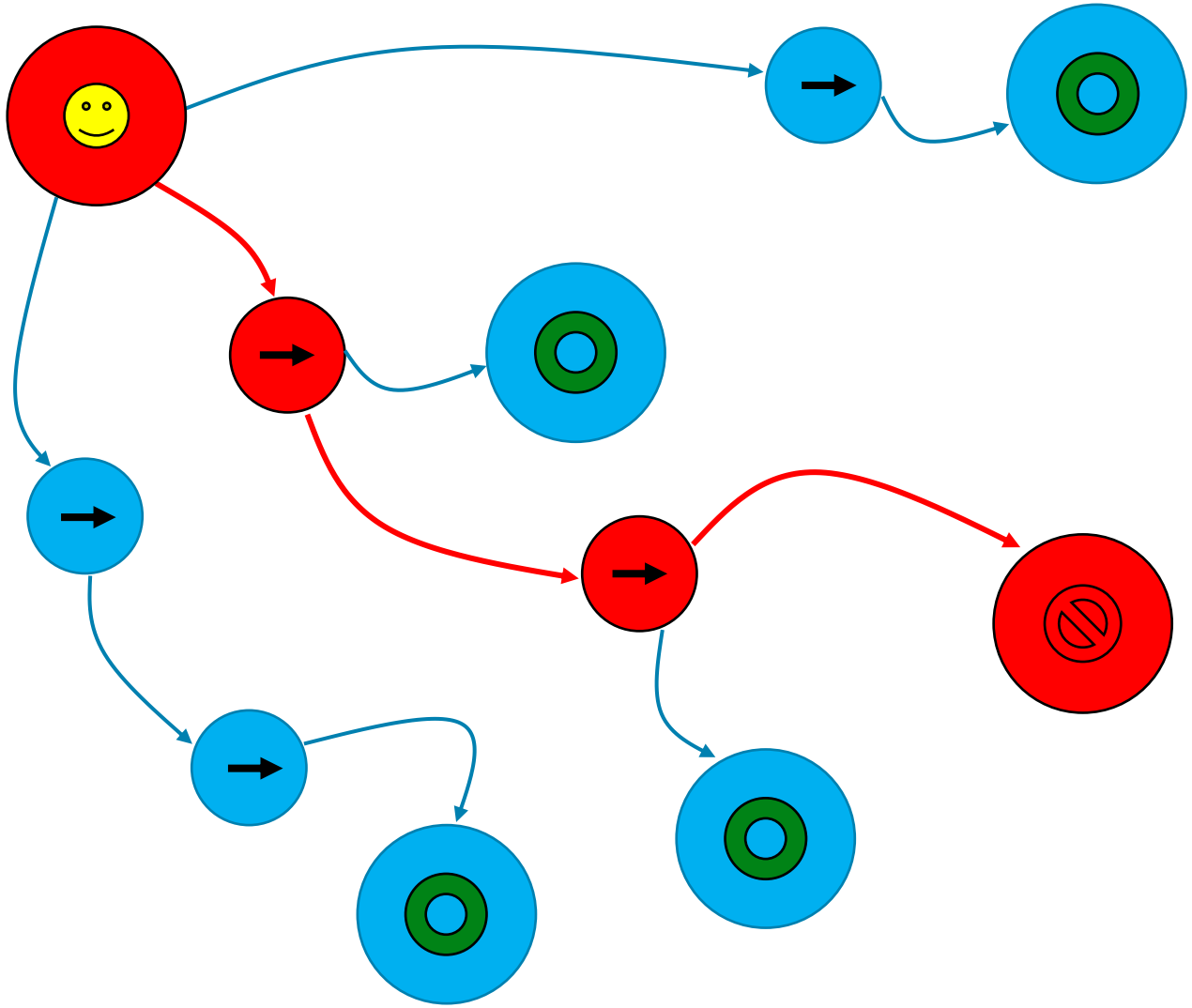
Taint analysis



Taint analysis



Taint analysis



Taint analysis

- Что нужно для анализа taint анализа?
  - Источники
  - Передатчики
  - Приёмники

Taint analysis: источники

- `HttpRequest.QueryString`
- `HttpRequest.Form`

Taint analysis: источники

- `HttpRequest.QueryString`
- `HttpRequest.Form`
- `TextBox.Text`

Taint analysis: источники

- `HttpRequest.QueryString`
- `HttpRequest.Form`
- `TextBox.Text`
- `Console.ReadLine`
- `gets`

Taint analysis: источники

- `HttpRequest.QueryString`
- `HttpRequest.Form`
- `TextBox.Text`
- `Console.ReadLine`
- `gets`
- `stdin`
  - `__acrt_iob_func(0)`
  - `&__iob_func()[0]`
  - `&_iob[0]`

Taint analysis: источники

- HttpRequest.QueryString
- HttpRequest.Form
- TextBox.Text
- Console.ReadLine
- gets
- stdin
  - __acrt_iob_func(0)
  - &__iob_func()[0]
  - &_iob[0]
- ....

SQLI

```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

SQLI

```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

Taint analysis: приёмники

Taint analysis: приёмники (sinks)

Taint analysis: приёмники (sinks)

- Специфичны для разных дефектов безопасности

Taint analysis: приёмники (sinks)

- Специфичны для разных дефектов безопасности
- SQLI:
  - параметр конструктора (команда)
  - св-во (команда)

Taint analysis: приёмники (sinks)

- Специфичны для разных дефектов безопасности
- SQLI:
  - параметр конструктора (команда)
  - св-во (команда)
- XSS:
  - `Response.Write`

Taint analysis: приёмники (sinks)

- Специфичны для разных дефектов безопасности
- SQLI:
  - параметр конструктора (команда)
  - св-во (команда)
- XSS:
  - `Response.Write`
- Path traversal:
  - Файловые операции

Taint analysis: приёмники (sinks)

```
var taintedStr = GetTaintedData();
var sqlCommand = new SqlCommand(taintedStr);
```

Taint analysis: приёмники (sinks)

```
var taintedStr = GetTaintedData();
var sqlCommand = new SqlCommand(taintedStr);
```

SQLI

```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

SQLI

```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

Taint analysis: распространение заражения

Taint analysis: распространение заражения

- Простые присваивания
- Функции
- Методы
- Индексаторы
- Конкатенация
- Интерполяция
- И т.п.

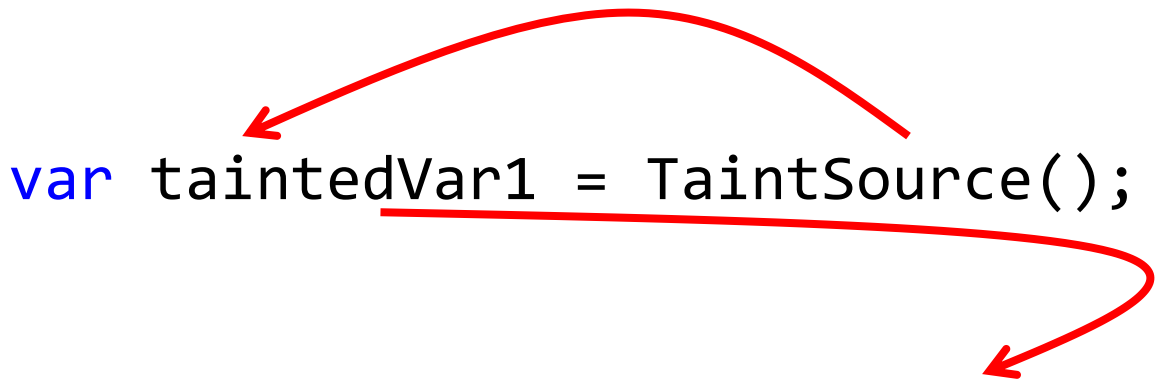
Taint analysis: заражение

```
var taintedVar1 = TaintSource();

var taintedVar2 = taintedVar1;
```

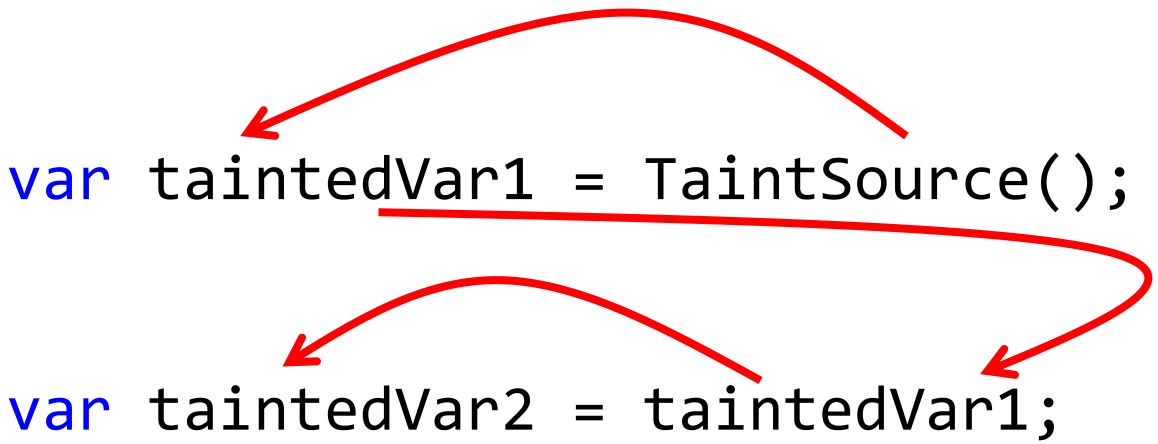
Taint analysis: заражение

```
var taintedVar1 = TaintSource();

var taintedVar2 = taintedVar1;
```

Taint analysis: заражение



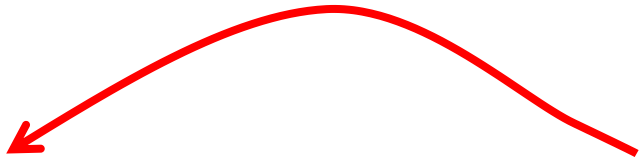
Taint analysis: заражение



Taint analysis: заражение

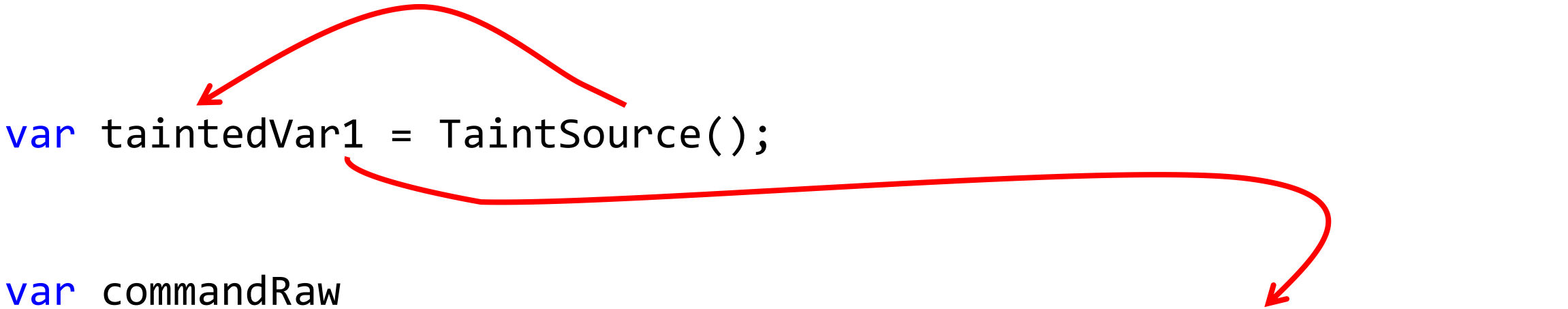
```
var taintedVar1 = TaintSource();


var commandRaw
  = "SELECT * FROM Users WHERE UserName = '" + taintedVar1 + "'";
```

Taint analysis: заражение



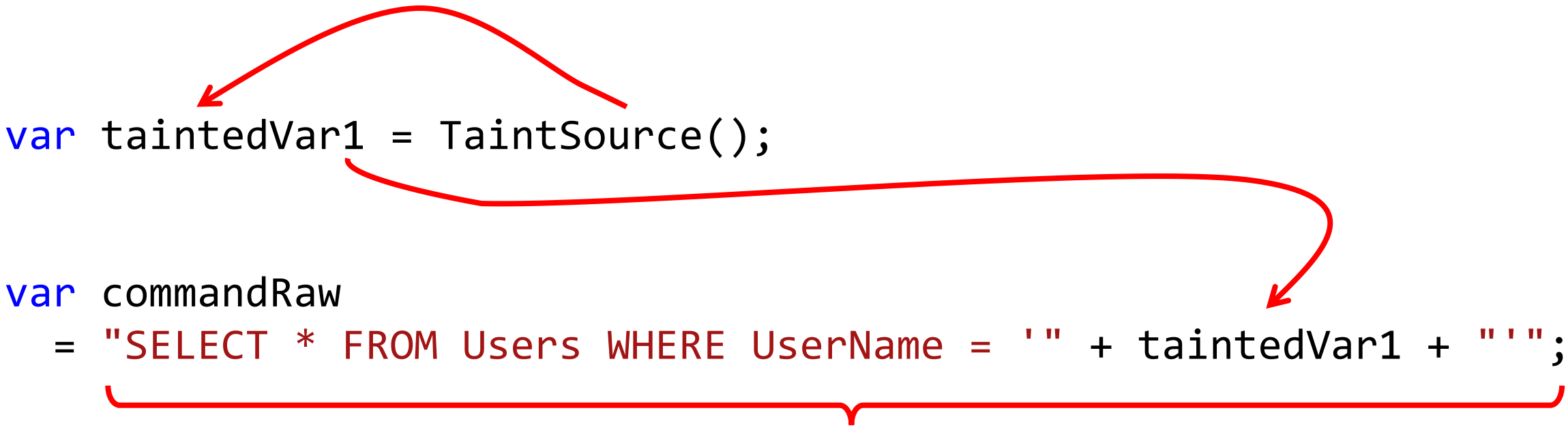
```
var taintedVar1 = TaintSource();


var commandRaw
  = "SELECT * FROM Users WHERE UserName = '" + taintedVar1 + "'";
```

Taint analysis: заражение



```
var taintedVar1 = TaintSource();

var commandRaw
  = "SELECT * FROM Users WHERE UserName = '" + taintedVar1 + "'";
```

Taint analysis: заражение



Taint analysis: заражение

```
StringBuilder command = new StringBuilder();


var taintedVar = TaintSource();


command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);


var resultCommand = command.ToString();
```

Taint analysis: заражение

```
StringBuilder command = new StringBuilder();


var taintedVar = TaintSource();


command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);


var resultCommand = command.ToString();
```

Taint analysis: заражение

```
StringBuilder command = new StringBuilder();


var taintedVar = TaintSource();


command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);


var resultCommand = command.ToString();
```

Taint analysis: заражение

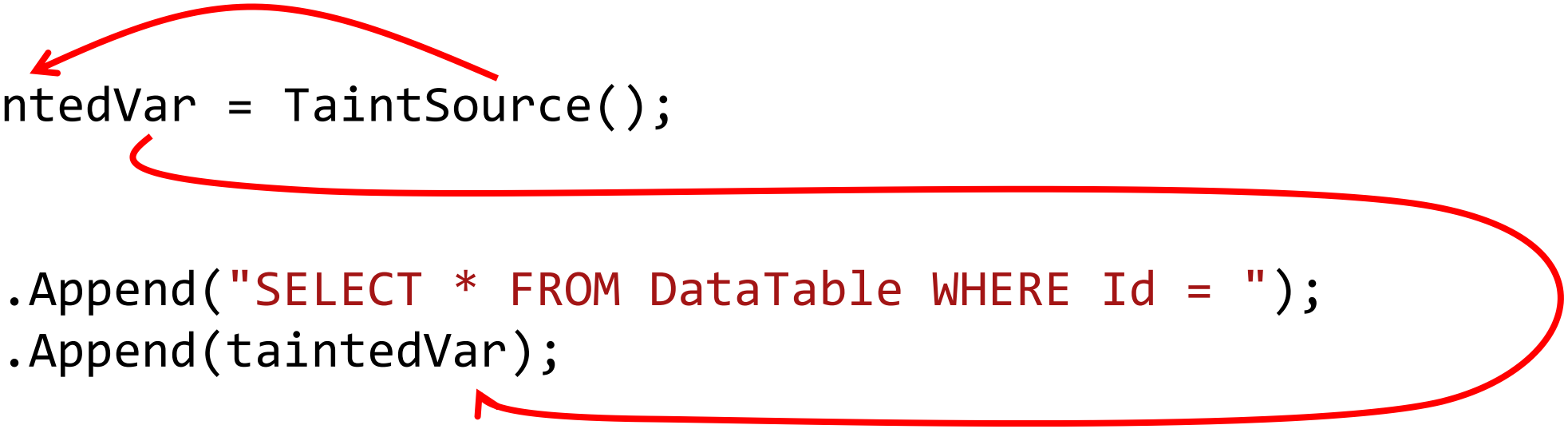
```
StringBuilder command = new StringBuilder();

var taintedVar = TaintSource();

command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);

var resultCommand = command.ToString();
```

Taint analysis: заражение

```
StringBuilder command = new StringBuilder();

var taintedVar = TaintSource();

command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);

var resultCommand = command.ToString();
```

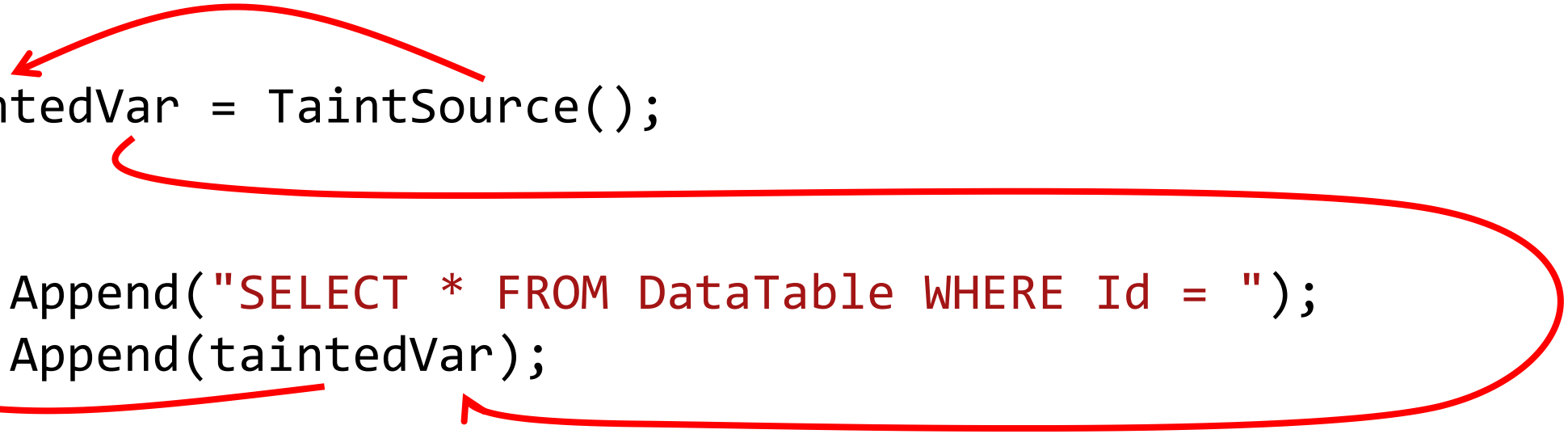
Taint analysis: заражение

```
StringBuilder command = new StringBuilder();

var taintedVar = TaintSource();

command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);

var resultCommand = command.ToString();
```



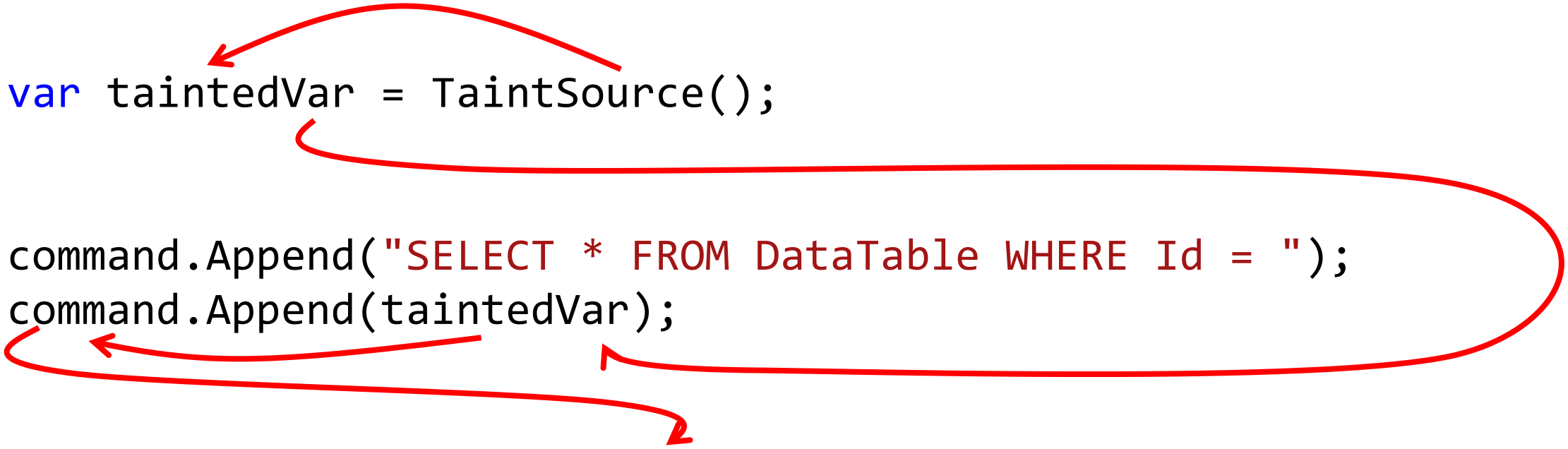
Taint analysis: заражение

```
StringBuilder command = new StringBuilder();

var taintedVar = TaintSource();

command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);

var resultCommand = command.ToString();
```



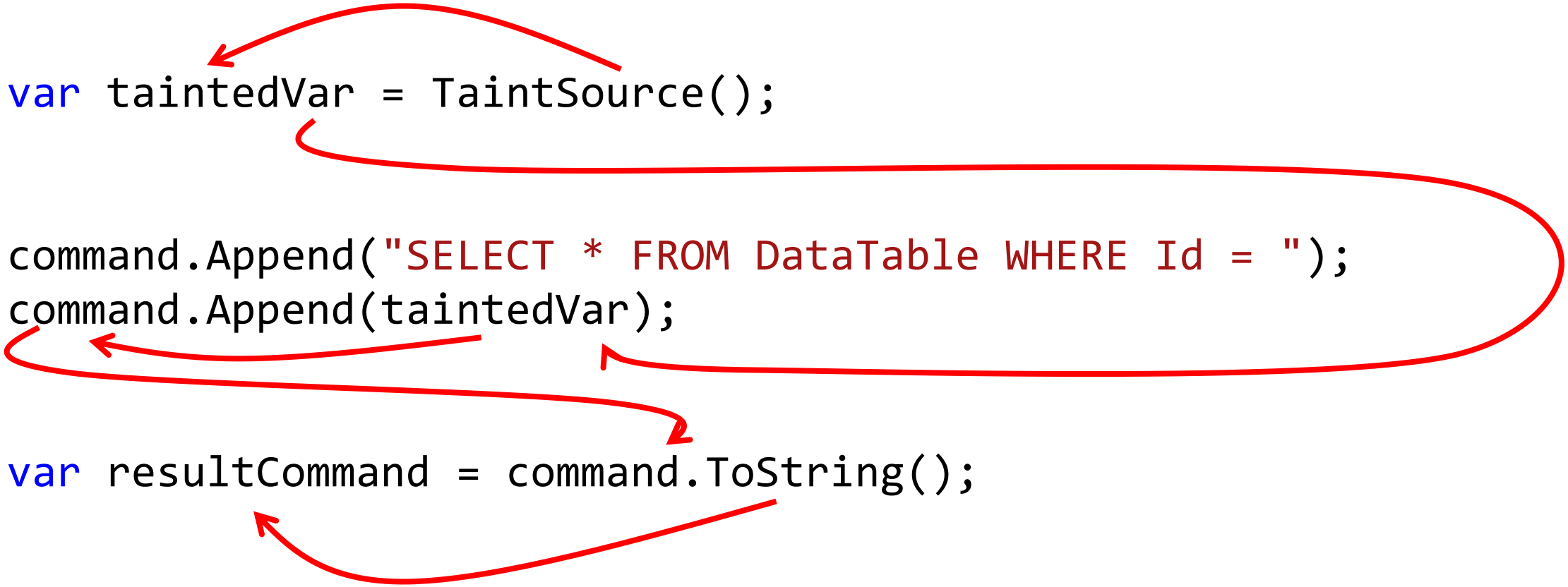
Taint analysis: заражение

```
StringBuilder command = new StringBuilder();

var taintedVar = TaintSource();

command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);

var resultCommand = command.ToString();
```



Taint analysis: заражение

```
StringBuilder command = new StringBuilder();

var taintedVar = TaintSource();

command.Append("SELECT * FROM DataTable WHERE Id = ");
command.Append(taintedVar);

var resultCommand = command.ToString();
```



Taint analysis: пример с NcFTP

libidn

```
else if (fgets (
            readbuf, BUFSIZ, stdin) == NULL) {
  ....
}

if (readbuf[strlen (readbuf) - 1] == '\n')
  readbuf[strlen (readbuf) - 1] = '\0';
```

NcFTP

```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

C:\Windows\System32\cmd.exe

```
D:\OSP\ncftp-3.2.6\ncftp\Release>
```

NcFTP

```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

NcFTP

```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

NcFTP

```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

NcFTP

```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

NcFTP

```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

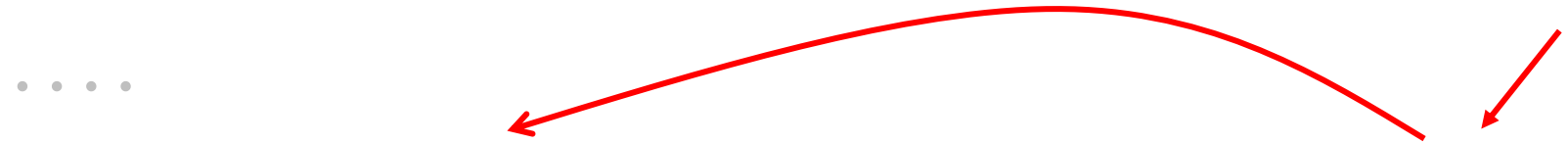
NcFTP software

NcFTP

```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

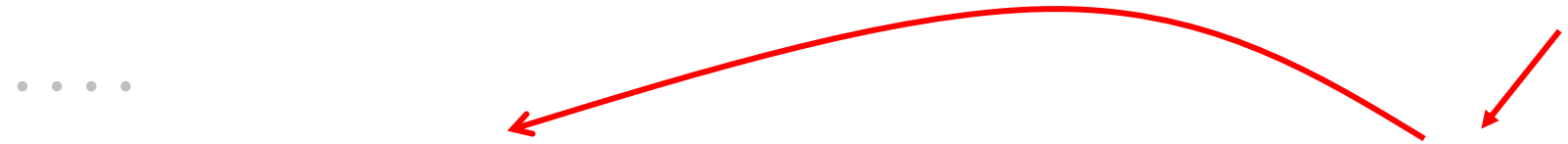
NcFTP



```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

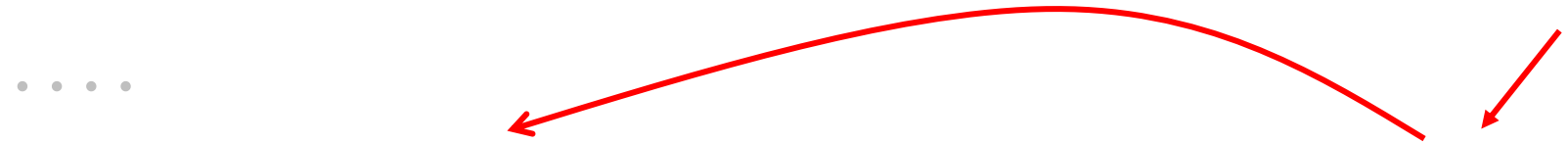
NcFTP



```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

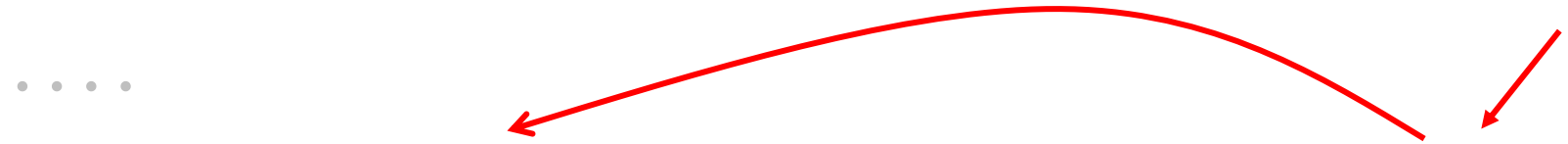
NcFTP



```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

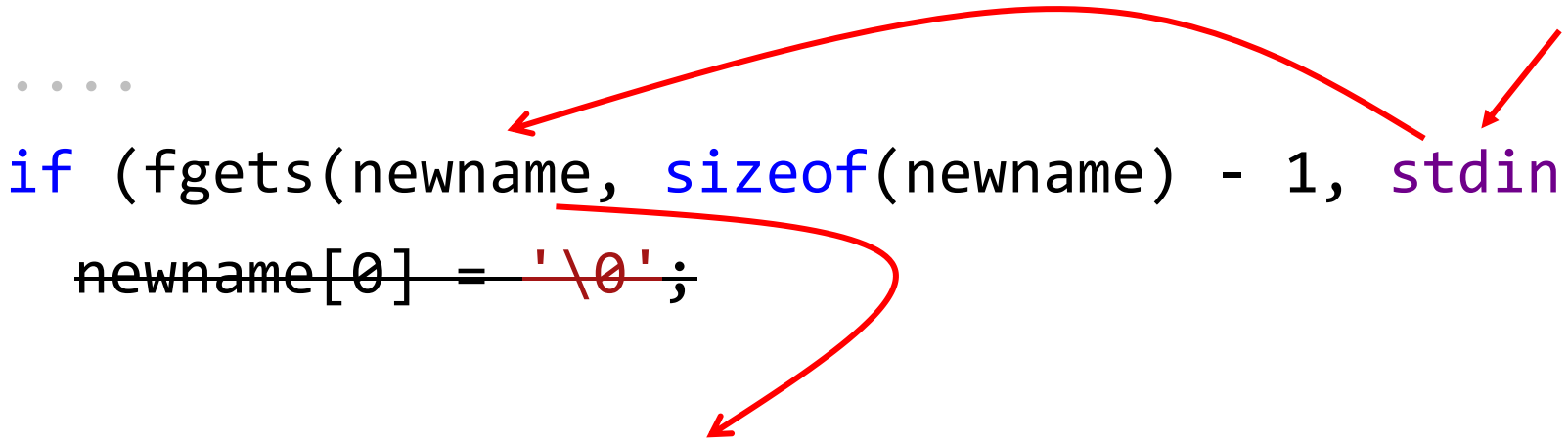
NcFTP



```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

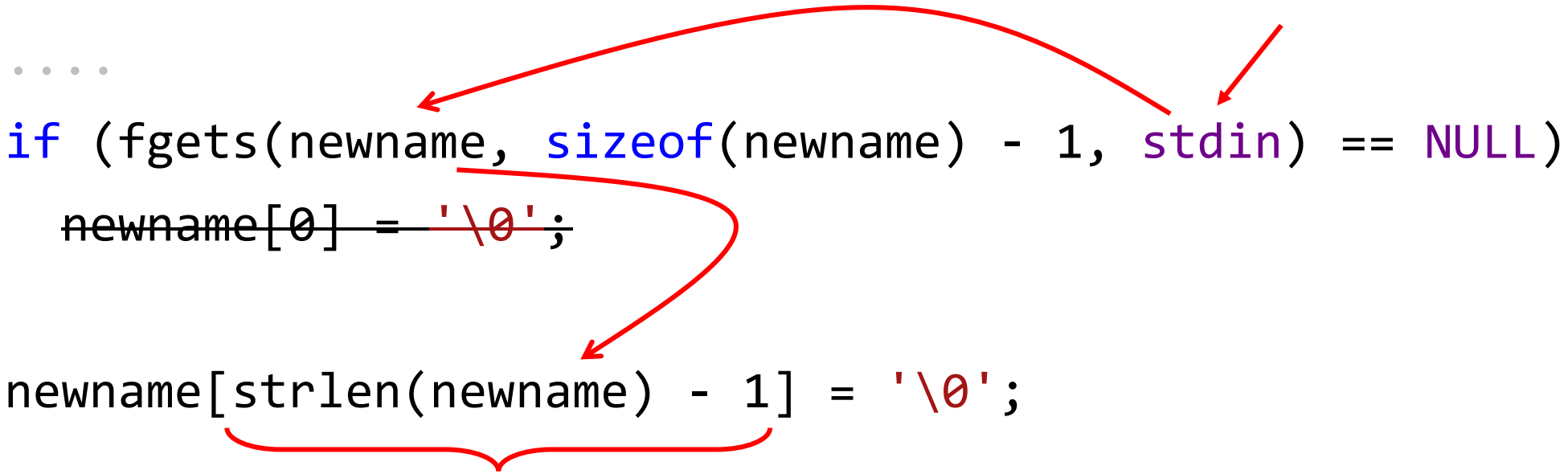
NcFTP



```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

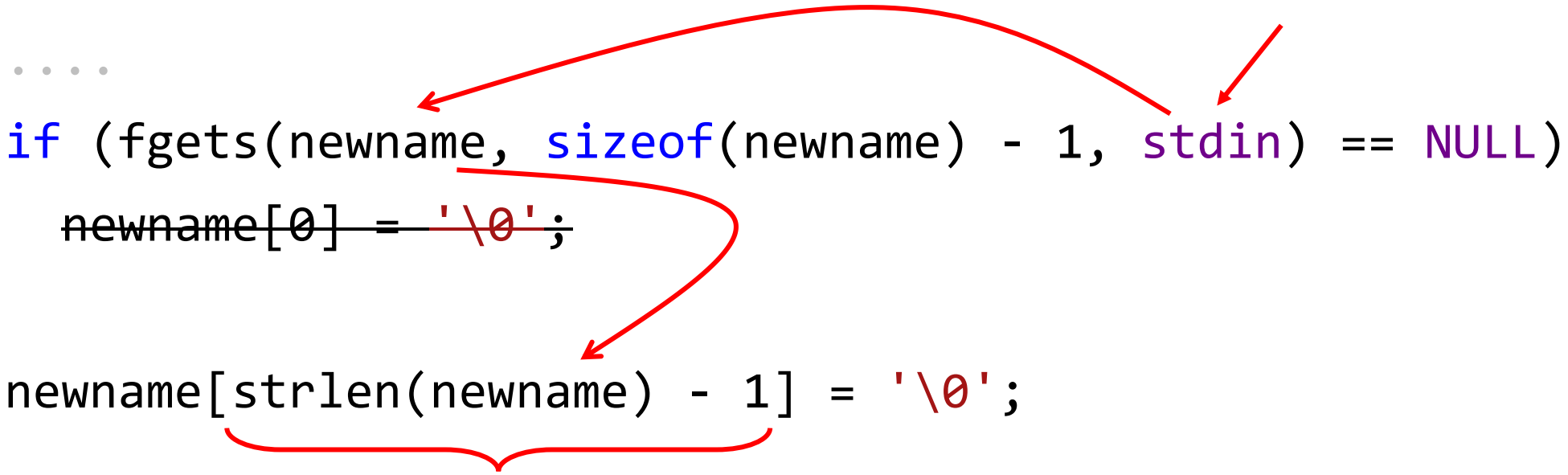
NcFTP



```
....
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
   newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

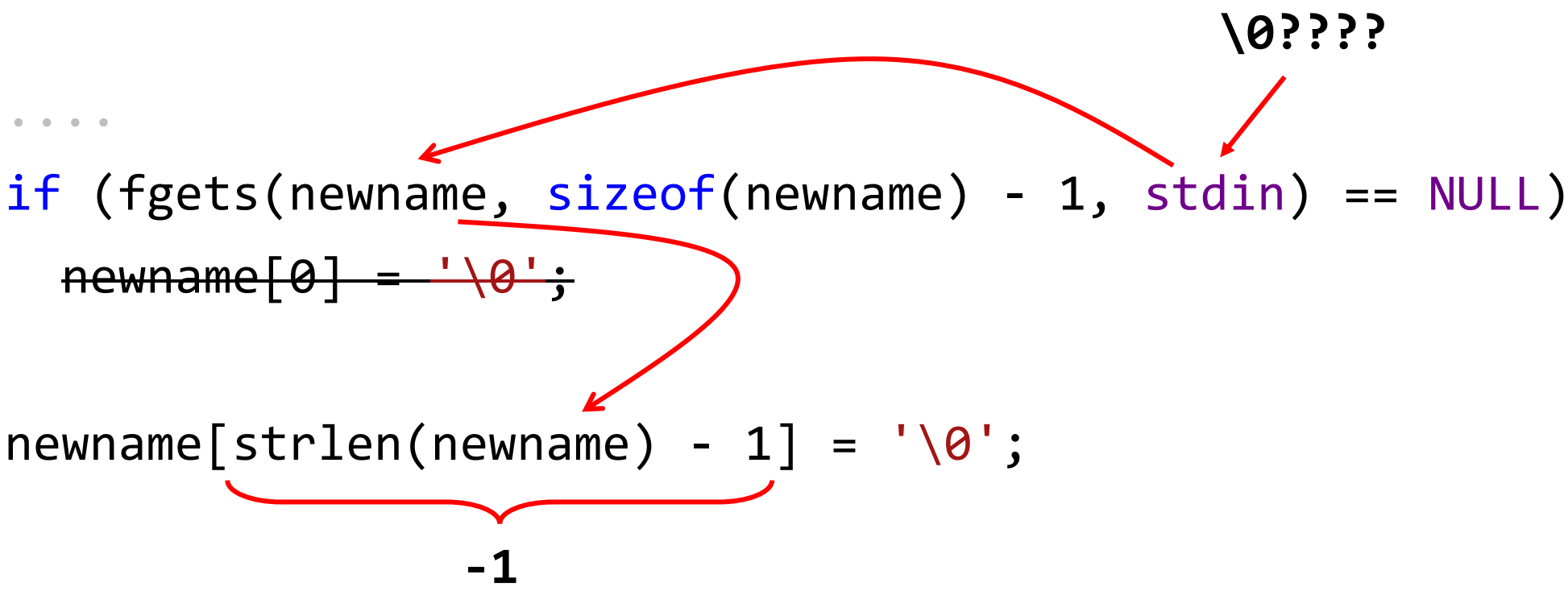
NcFTP



```
. . . .
if (fgets(newname, sizeof(newname) - 1, stdin) == NULL)
  newname[0] = '\0';

newname[strlen(newname) - 1] = '\0';
```

NcFTP software

NcFTP



NcFTP software

NcFTP

```
newname[-1] = '\0';
```

Taint analysis: пример с SQLI

SQLI

```
using (SqlConnection connection = new SqlConnection(_connectionString))
{
  String userName = Request.Form["userName"];
  using (var command = new SqlCommand()
  {
    Connection = connection,
    CommandText = $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
    CommandType = System.Data.CommandType.Text
  })
  {
    using (var reader = command.ExecuteReader())
      // Data processing
  }
}
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];
```

```
using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];


using (var command = new SqlCommand()
{
  Connection = connection,

  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",

  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI: защищаемся

SQLI

```
String userName = Request.Form["userName"];
using (var command = new SqlCommand()
{
  Connection = connection,
  CommandText =
    $"SELECT * FROM Users WHERE UserName = '" + userName + "'",
  CommandType = System.Data.CommandType.Text
})
  ....
```

SQLI

```
String userName = Request.Form["userName"];
using (var command = new SqlCommand()
{
  Connection = connection,
  CommandText =
    $"SELECT * FROM Users WHERE UserName = @userName",
  CommandType = System.Data.CommandType.Text
}) {
  var userNameParam = new SqlParameter("@userName", userName);
  command.Parameters.Add(userNameParam);
}
```

SQLI

```
String userName = Request.Form["userName"];
using (var command = new SqlCommand()
{
  Connection = connection,
  CommandText =
    $"SELECT * FROM Users WHERE UserName = @userName",
  CommandType = System.Data.CommandType.Text
}) {
  var userNameParam = new SqlParameter("@userName", userName);
  command.Parameters.Add(userNameParam);
}
```

SQLI

```
String userName = Request.Form["userName"];
using (var command = new SqlCommand()
{
  Connection = connection,
  CommandText =
    $"SELECT * FROM Users WHERE UserName = @userName",
  CommandType = System.Data.CommandType.Text
}) {
  var userNameParam = new SqlParameter("@userName", userName);
  command.Parameters.Add(userNameParam);
}
```

SQLI

```
String userName = Request.Form["userName"];
using (var command = new SqlCommand()
{
  Connection = connection,
  CommandText =
    $"SELECT * FROM Users WHERE UserName = @userName",
  CommandType = System.Data.CommandType.Text
}) {
  var userNameParam = new SqlParameter("@userName", userName);
  command.Parameters.Add(userNameParam);
}
```

SQLI

```
String userName = Request.Form["userName"];
using (var command = new SqlCommand()
{
  Connection = connection,
  CommandText =
    $"SELECT * FROM Users WHERE UserName = @userName",
  CommandType = System.Data.CommandType.Text
}) {
  var userNameParam = new SqlParameter("@userName", userName);
  command.Parameters.Add(userNameParam);
}
```

SQLI

```
String userName = Request.Form["userName"];
using (var command = new SqlCommand()
{
  Connection = connection,
  CommandText =
    $"SELECT * FROM Users WHERE UserName = @userName",
  CommandType = System.Data.CommandType.Text
}) {
  var userNameParam = new SqlParameter("@userName", userName);
  command.Parameters.Add(userNameParam);
}
```

Taint analysis: ограничения

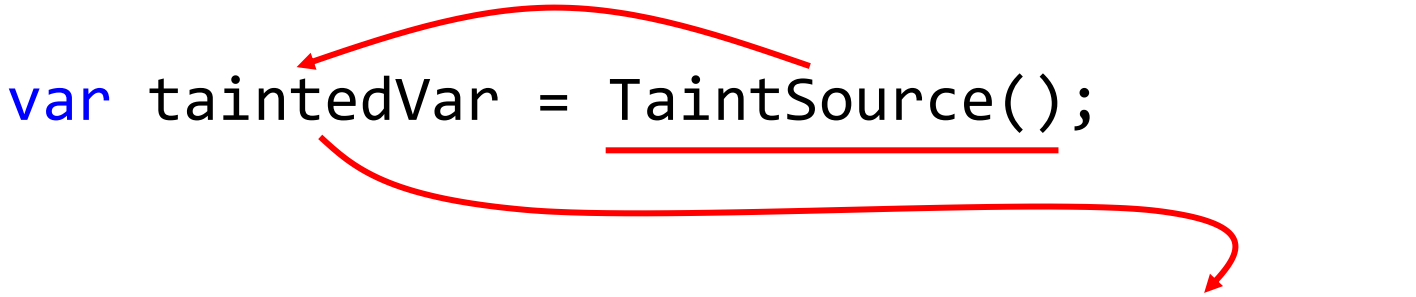
Taint analysis: ограничения

```
var taintedVar = TaintSource();

var anotherTaintedVar = Foo(taintedVar);

TaintSink(anotherTaintedVar);
```

Taint analysis: ограничения

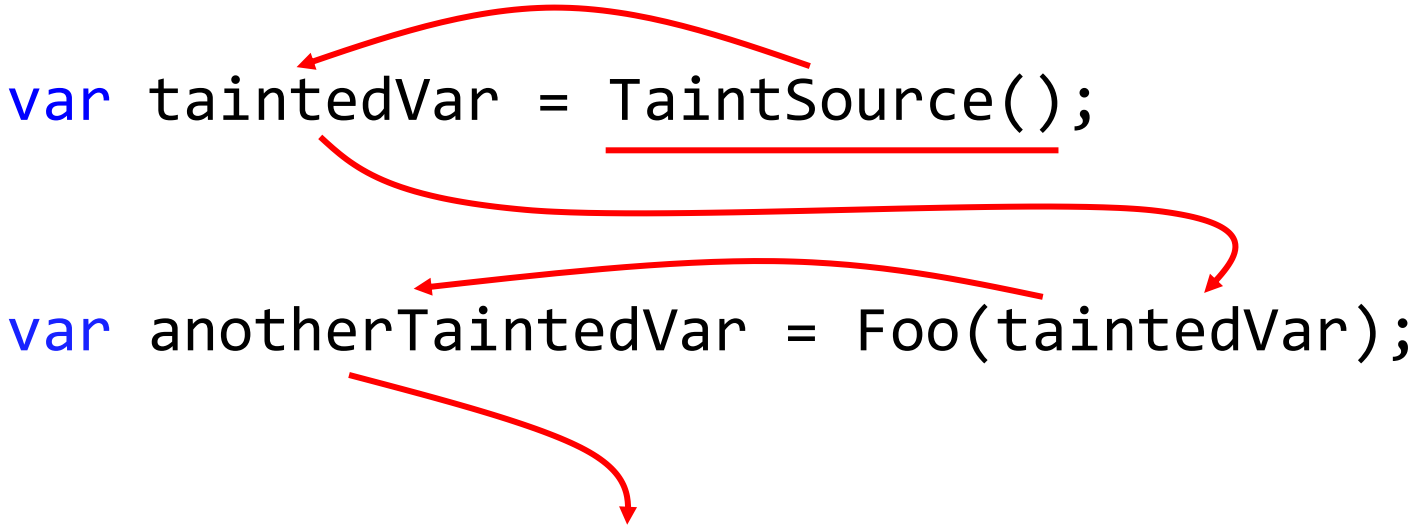
```
var taintedVar = TaintSource();

var anotherTaintedVar = Foo(taintedVar);

TaintSink(anotherTaintedVar);
```

Taint analysis: ограничения

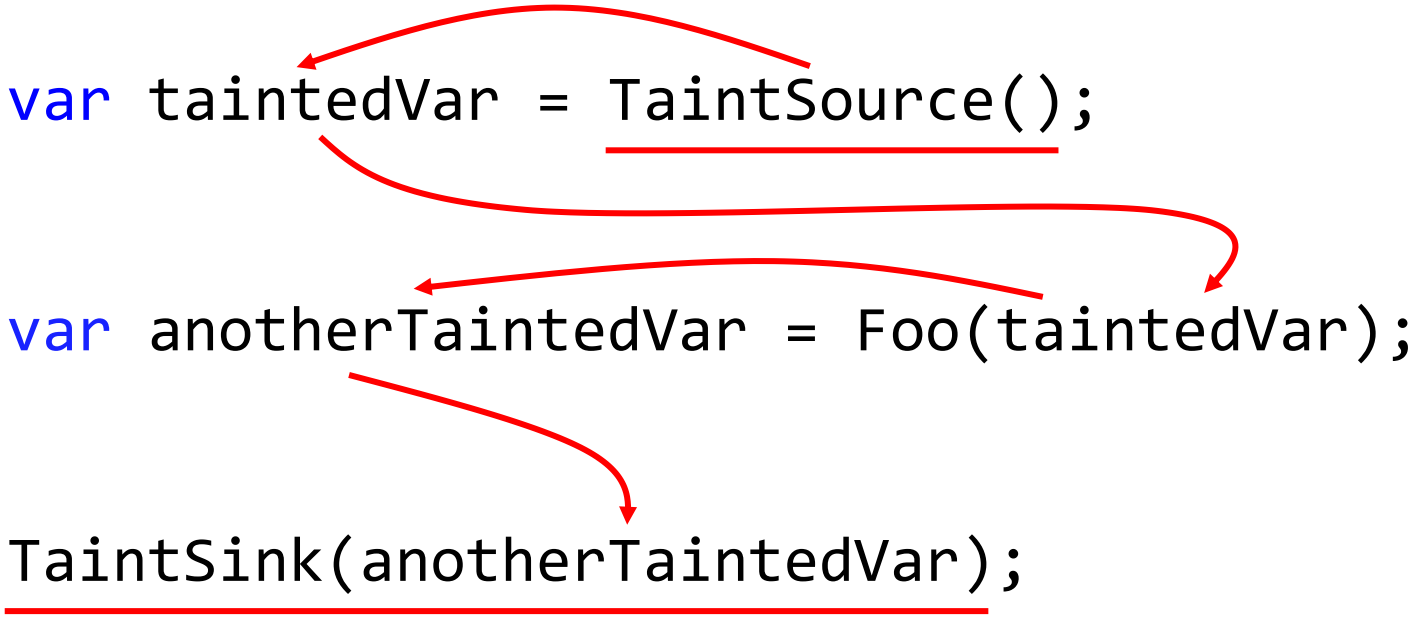


```
var taintedVar = TaintSource();

var anotherTaintedVar = Foo(taintedVar);


TaintSink(anotherTaintedVar);
```

Taint analysis: ограничения



Taint analysis: ограничения



Закрепление: taint analysis

- Решает проблему излишнего доверия к внешним данным
- Отслеживает распространение 'заражённых' данных по приложению
- Хорошо подходит для поиска различного рода инъекций

- Основные ограничения – отсутствие информации о:
  - источниках;
  - приёмниках;
  - разрывы трасс передачи данных.

SAST

- SAST – не панацея

SAST

- SAST – не панацея
- Да здравствует SAST!
- Эффективно сочетается с другими методологиями

За безопасность необходимо *платить*,

а за ее отсутствие – *расплачиваться.*

Q&A

pvs-studio.com
vasiliev@viva64.com