

# ВОКРУГ РБПО ЗА 25 ВЕБИНАРОВ

ГОСТ Р 56939-2024

## Вебинар 10. Статический анализ исходного кода

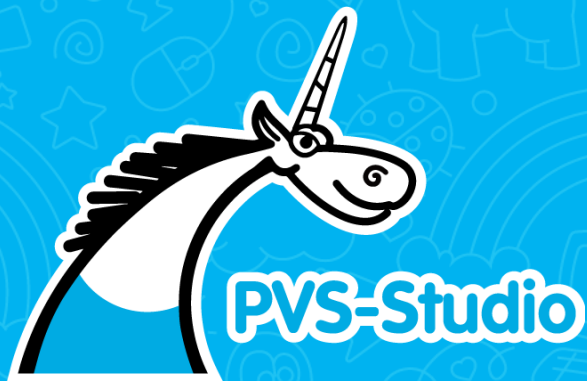


Что такое статический анализ

SAST: безопасность и  
защищенность

Особенности внедрения  
инструмента

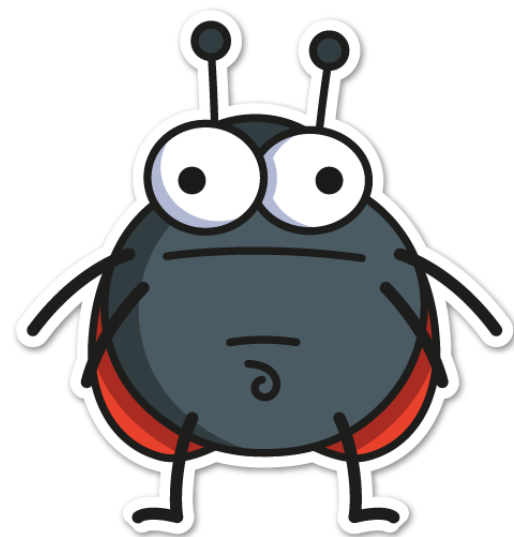
# Зачем нужны инструменты статического анализа





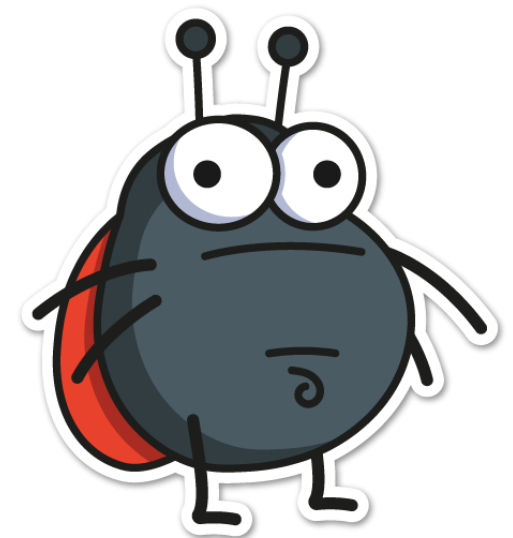
# Проблематика

- Объем кодовой базы растёт, её становится тяжелее контролировать



# Проблематика

- Объем кодовой базы растёт, её становится тяжелее контролировать
- Плотность ошибок начинает расти нелинейно



# Проблематика

- Объем кодовой базы растёт, её становится тяжелее контролировать
- Плотность ошибок начинает расти нелинейно
- Старых методов контроля качества уже недостаточно



# Проблематика

- Объем кодовой базы растёт, её становится тяжелее контролировать
- Плотность ошибок начинает расти нелинейно
- Старых методов контроля качества уже недостаточно
- Появляется больше разных ошибок
  - Которые ловят тесты / не ловят тесты
  - Опасные / некритичные



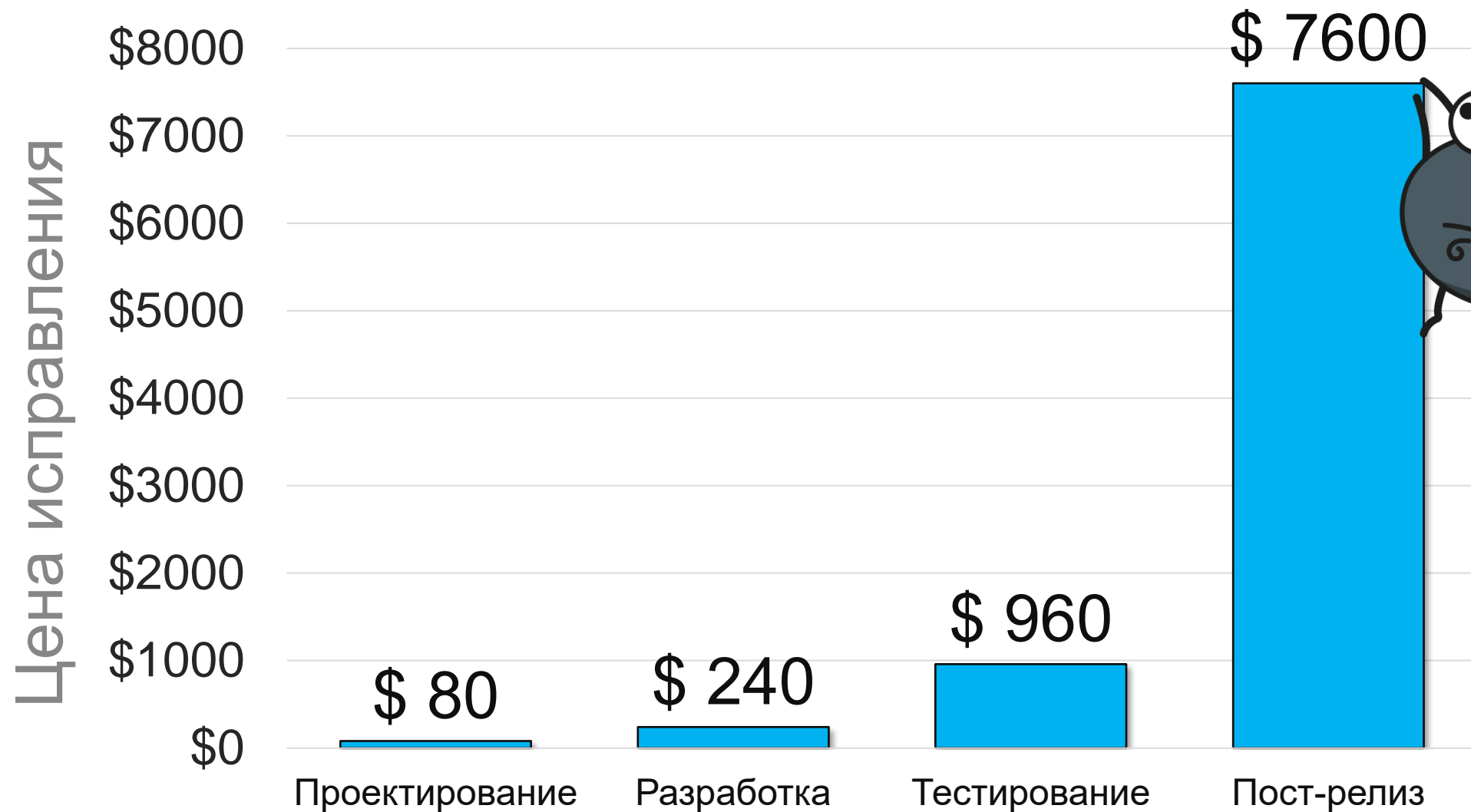
# Проблематика

- Объем кодовой базы растёт, её становится тяжелее контролировать
- Плотность ошибок начинает расти нелинейно
- Старых методов контроля качества уже недостаточно
- Появляется больше разных ошибок
  - Которые ловят тесты / не ловят тесты
  - Опасные / некритичные
- ***И это еще не все...***



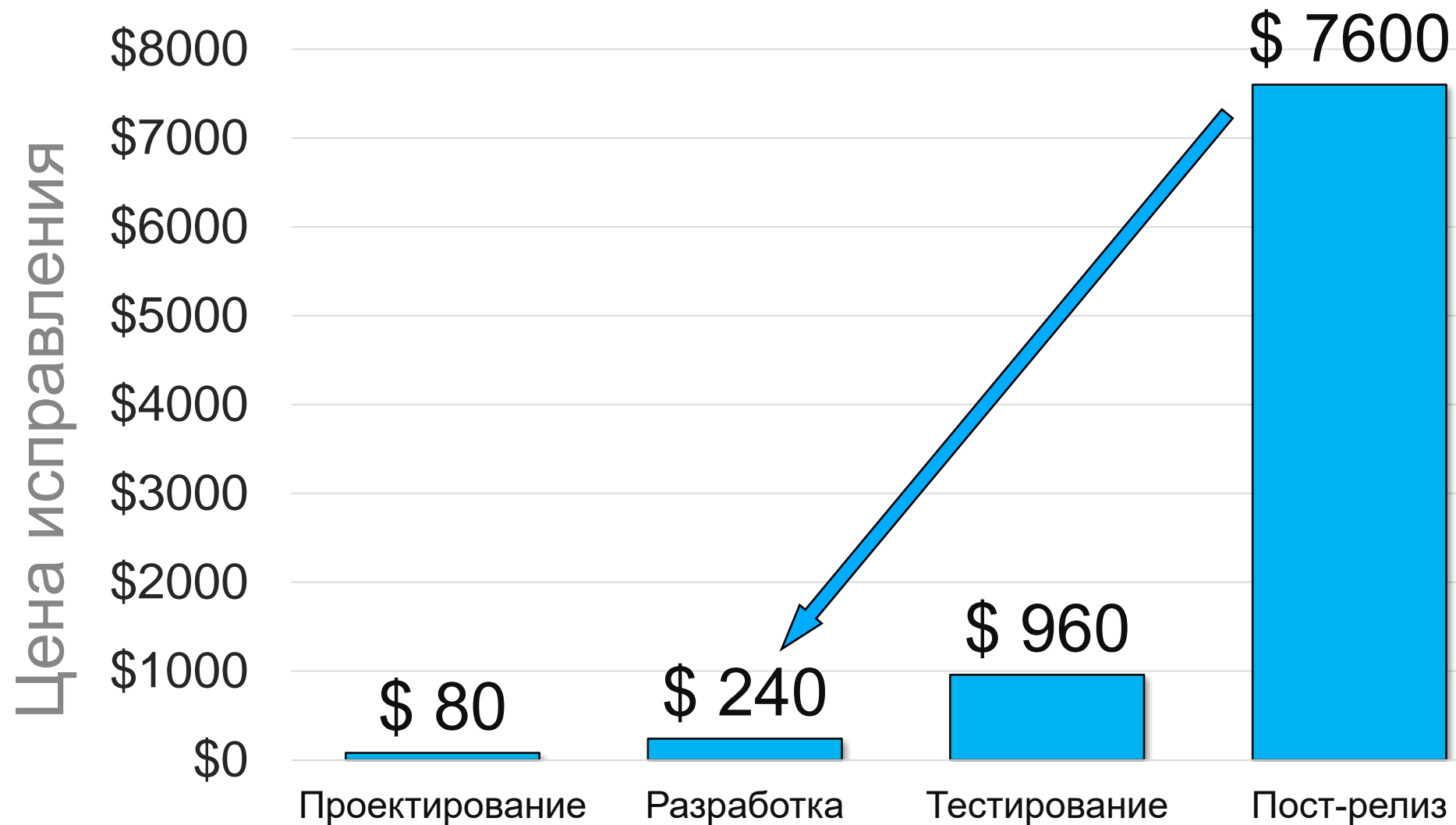


# Сколько стоит исправить уязвимость?



Источник - NIST: National Institute of Standards and Technology

# Сколько стоит исправить уязвимость?



Источник - NIST: National Institute of Standards and Technology

# Способы поиска ошибок

- Модульное тестирование
- Интеграционное тестирование
- Системное тестирование
- ....
- Динамический анализ
- Статический анализ



# Способы поиска ошибок

- Модульное тестирование
- Интеграционное тестирование
- Системное тестирование
- ....
- **Динамический анализ**
- **Статический анализ**

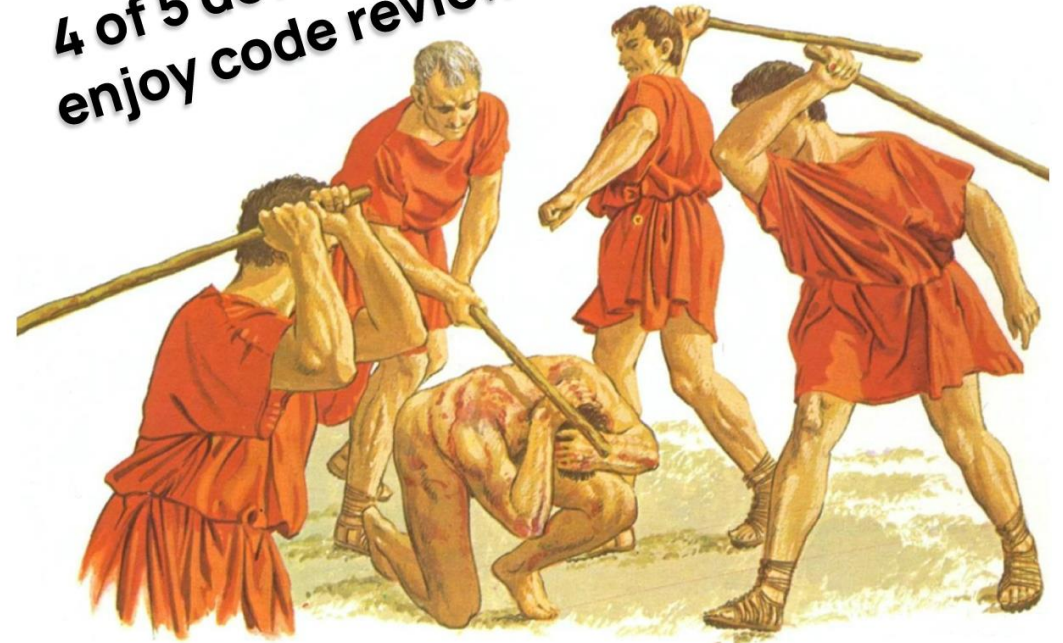




# Статический анализ

- В начале был код-ревью

*4 of 5 developers  
enjoy code review*

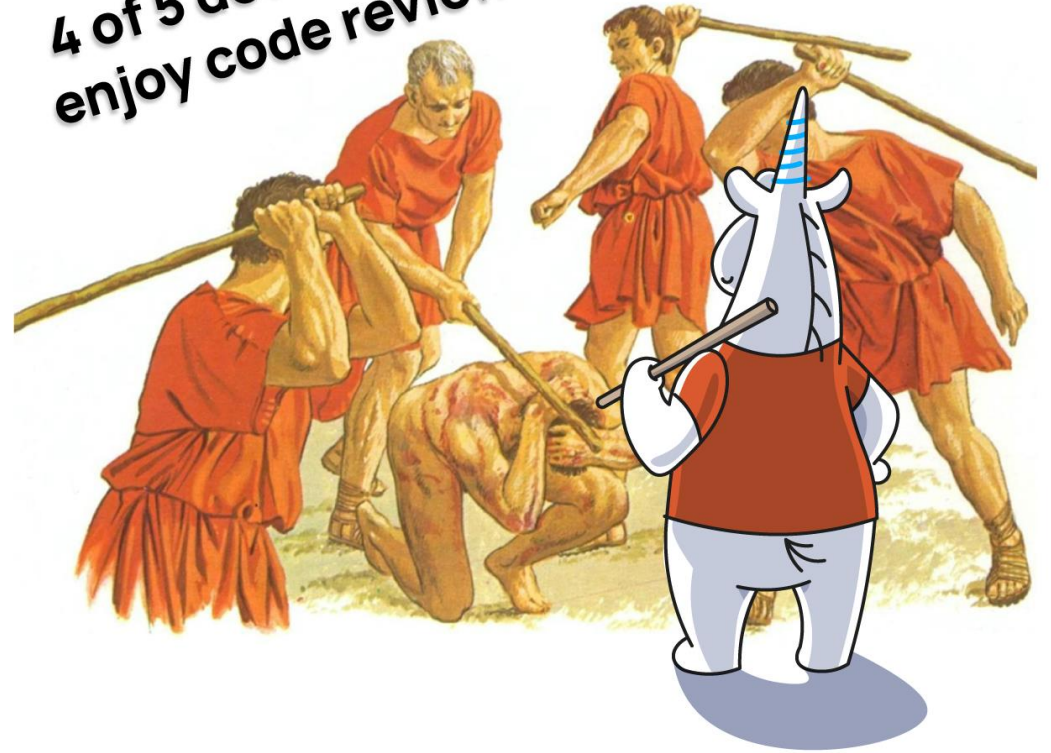




# Статический анализ

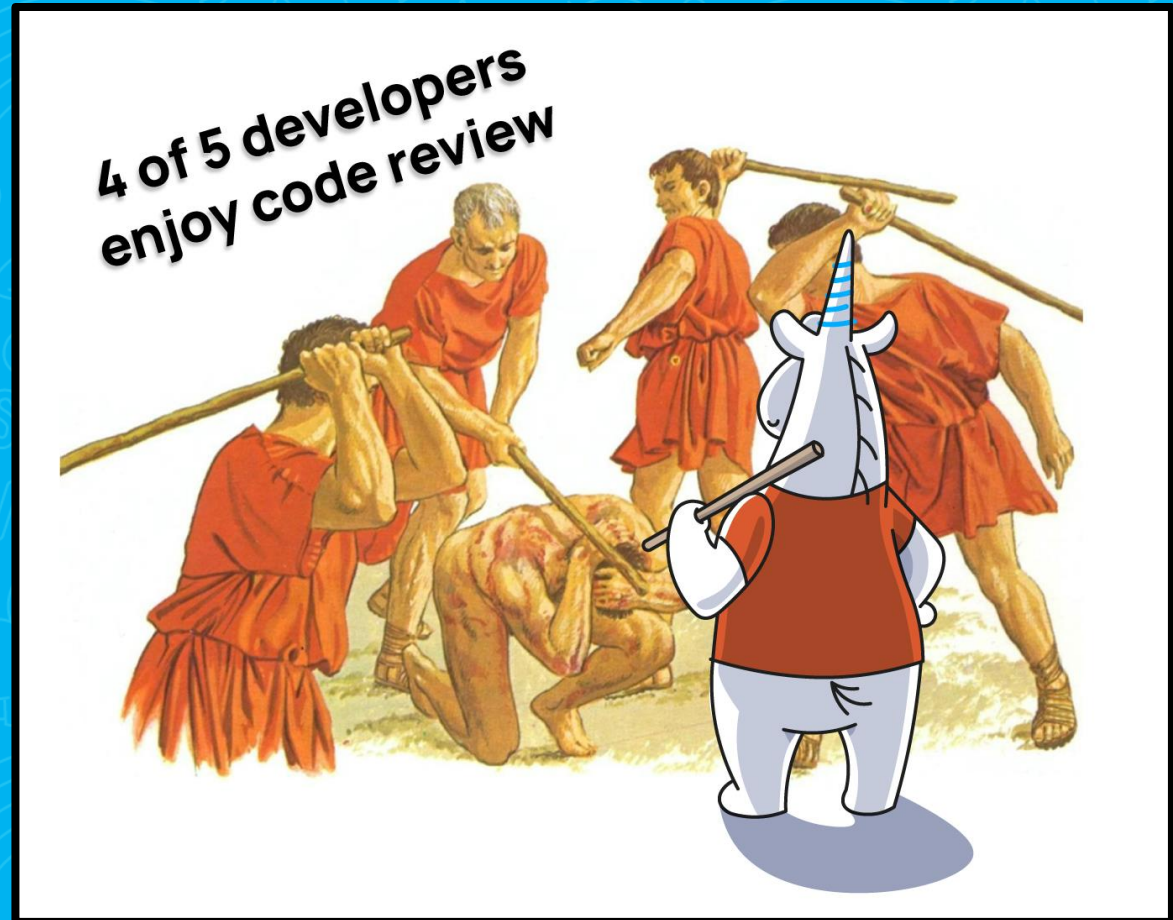
- Автоматический код-ревью!

4 of 5 developers  
enjoy code review



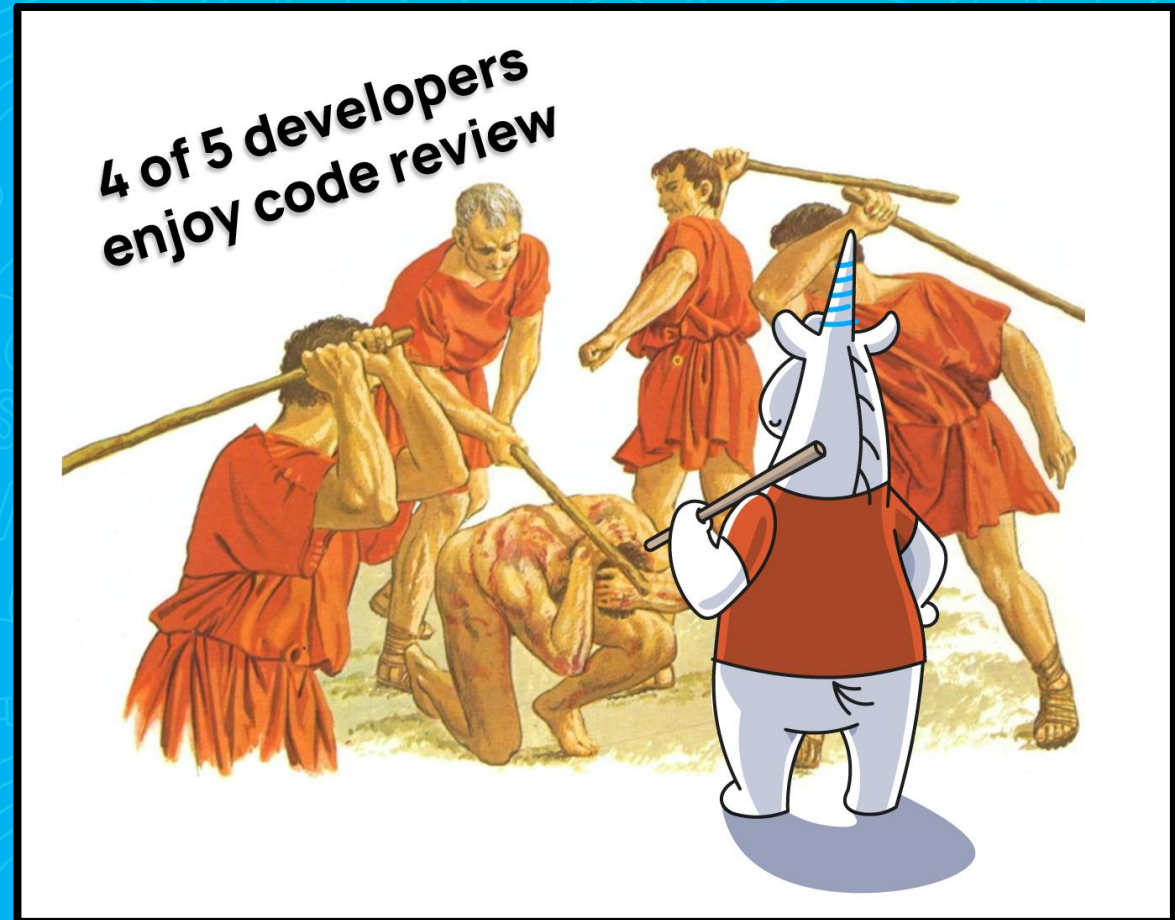
# Статический анализ

- Автоматический код-ревью!
- Нужен только код программы



# Статический анализ

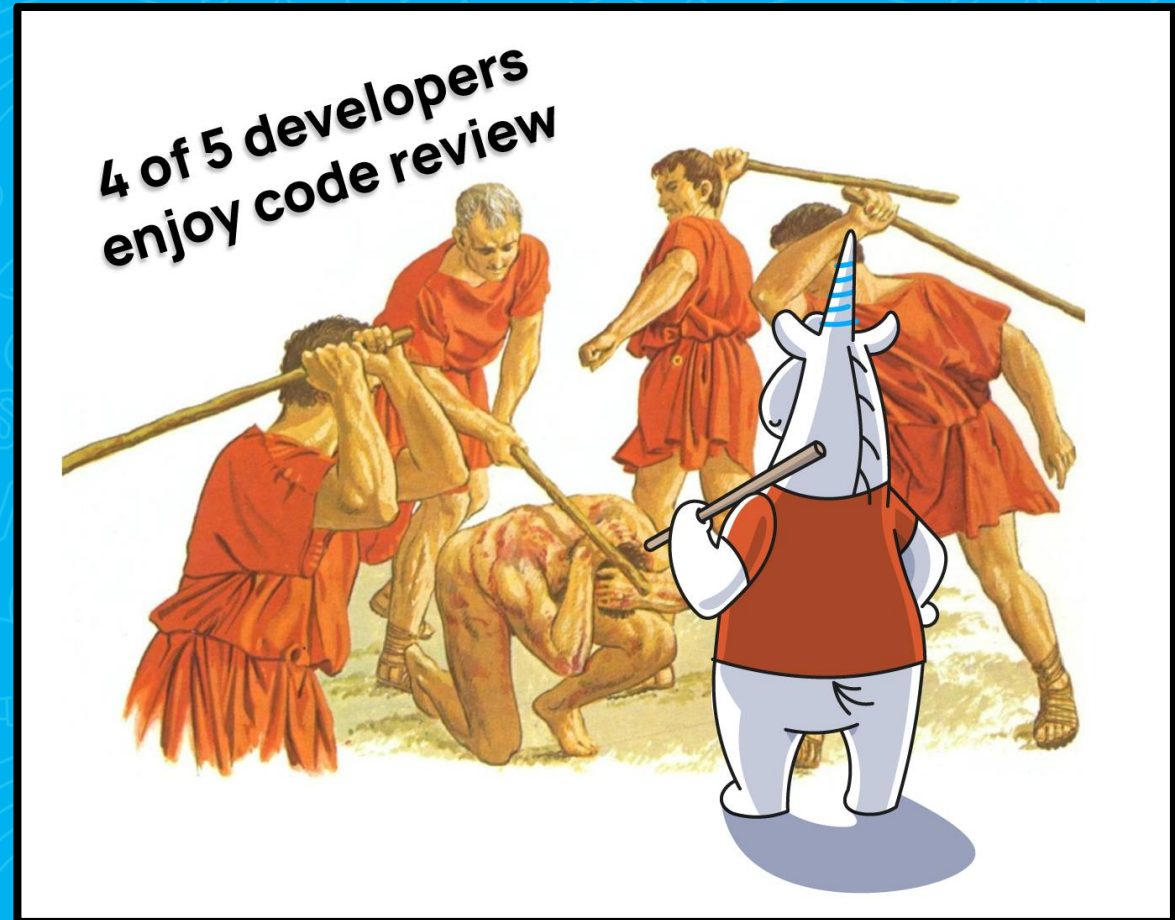
- Автоматический код-ревью!
- Нужен только код программы
- Полное покрытие проекта





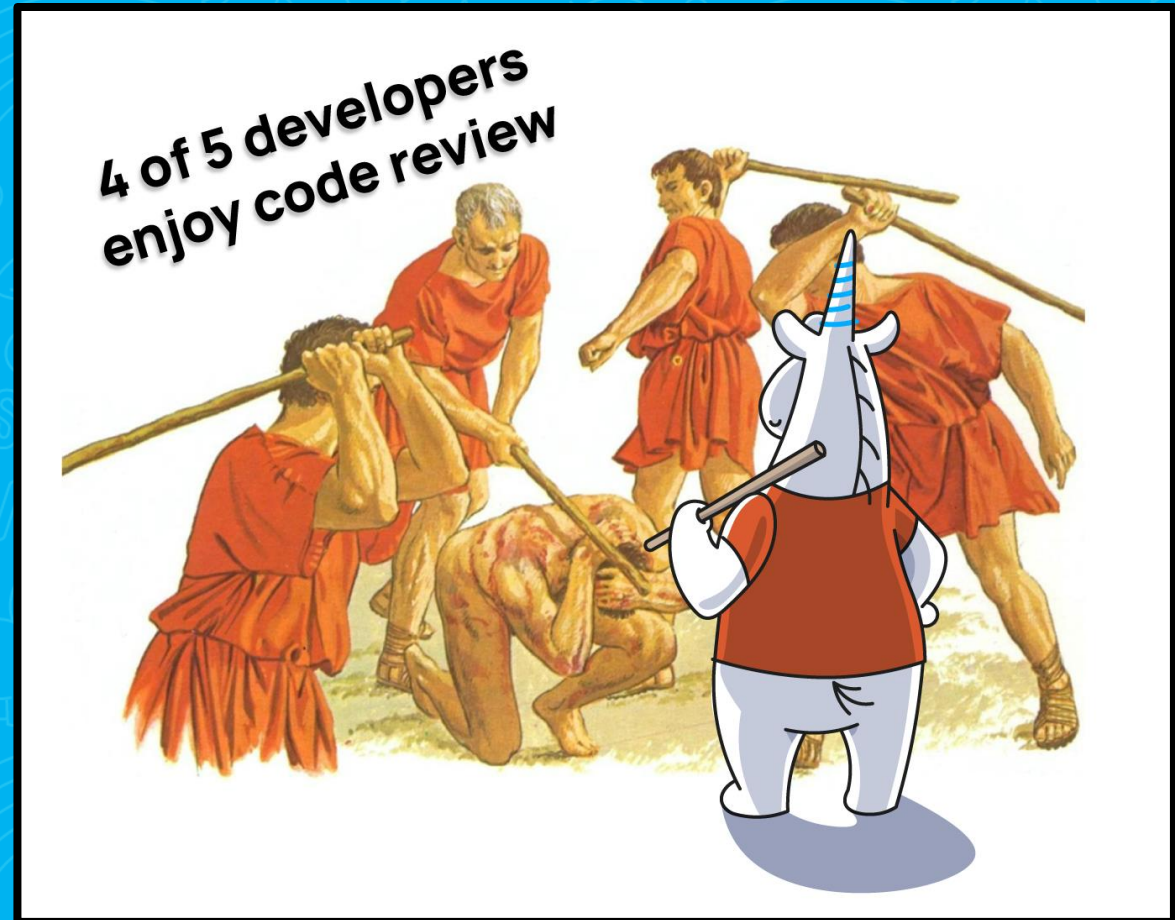
# Статический анализ

- Автоматический код-ревью!
- Нужен только код программы
- Полное покрытие проекта
- Раннее обнаружение ошибок



# Статический анализ

- Автоматический код-ревью!
- Нужен только код программы
- Полное покрытие проекта
- Раннее обнаружение ошибок
- Ошибки исправляются еще на этапе разработки





# Виды проблем

проблемы  
безопасности

неправильная работа  
с методами

недостижимый  
код

ошибки доступа к  
памяти

опечатки



ошибки сериализации /  
десериализации

выход за  
границы

ошибки  
синхронизации

переполнение  
буфера

неправильная работа с  
типами

# Ошибки которые сложно заметить

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```



ONLYOFFICE

# Ошибки которые сложно заметить

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```



ONLYOFFICE

# Ошибки которые сложно заметить



ONLYOFFICE

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```

# Ошибки которые сложно заметить

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```



ONLYOFFICE



# Ошибки которые сложно заметить

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```



ONLYOFFICE

# Ошибки которые сложно заметить



ONLYOFFICE

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```

# Ошибки которые сложно заметить

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```



ONLYOFFICE



# Ошибки которые сложно заметить

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```



ONLYOFFICE

# Ошибки которые сложно заметить



ONLYOFFICE

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsUserName = userName;
    CredentialsUserPassword = password;
    CredentialsDomain = domain;
}
```

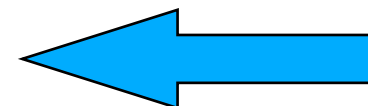
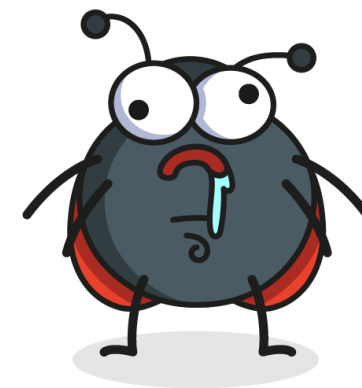


# Ошибки которые сложно заметить

```
public void SetCredentials
(string userName, string password, string domain) {
    if (string.IsNullOrEmpty(userName)) {
        throw new ArgumentException
            ("Empty user name.", "userName");
    }
    if (string.IsNullOrEmpty("password")) {
        throw new ArgumentException
            ("Empty password.", "password");
    }
    CredentialsDomain = domain;
}
```



ONLYOFFICE



## Предупреждение PVS-Studio:

V3022 Expression 'string.IsNullOrEmpty("password")' is always false.

# Ошибки которые не хочется искать

```
public struct BlobSasBuilder : IEquatable<BlobSasBuilder> {  
    ....  
    public bool Equals(BlobSasBuilder other) =>  
        BlobName == other.BlobName &&  
        CacheControl == other.CacheControl &&  
        BlobContainerName == other.BlobContainerName &&  
        ContentDisposition == other.ContentDisposition &&  
        ContentEncoding == other.ContentEncoding &&  
        ContentLanguage == other.ContentEncoding &&  
        ContentType == other.ContentType &&  
        ExpiryTime == other.ExpiryTime &&  
        Identifier == other.Identifier &&  
        IPRange == other.IPRange &&  
        Permissions == other.Permissions &&  
        Protocol == other.Protocol &&  
        StartTime == other.StartTime &&  
        Version == other.Version;  
}
```

Azure  
SDK

# Ошибки которые не хочется искать

31

Azure  
SDK

```
public struct BlobSasBuilder : IEquatable<BlobSasBuilder> {
```

```
....
```

```
public bool Equals(BlobSasBuilder other) =>
```

```
    BlobName == other.BlobName &&
```

```
    CacheControl == other.CacheControl &&
```

```
    BlobContainerName == other.BlobContainerName &&
```

```
    ContentDisposition == other.ContentDisposition &&
```

```
ContentEncoding == other.ContentEncoding &&
```

```
ContentLanguage == other.ContentEncoding &&
```

```
    ContentType == other.ContentType &&
```

```
    ExpiryTime == other.ExpiryTime &&
```

```
    Identifier == other.Identifier &&
```

```
    IPRange == other.IPRange &&
```

```
    Permissions == other.Permissions &&
```

```
    Protocol == other.Protocol &&
```

```
    StartTime == other.StartTime &&
```

```
    Version == other.Version;
```

# Ошибки которые не хочется искать

Azure  
SDK

```
public struct BlobSasBuilder : IEquatable<BlobSasBuilder> {
```

```
....
```

```
public bool Equals(BlobSasBuilder other) =>
```

```
    BlobName == other.BlobName &&
```

```
    CacheControl == other.CacheControl &&
```

```
    BlobContainerName == other.BlobContainerName &&
```

```
    ContentDisposition == other.ContentDisposition &&
```

```
    ContentEncoding == other.ContentEncoding &&
```

```
    ContentLanguage == other.ContentEncoding &&
```

```
    ContentTypes == other.ContentTypes &&
```

## Предупреждение PVS-Studio:

V3112 An abnormality within similar comparisons. It is possible that a typo is present inside the expression 'ContentLanguage == other.ContentEncoding'

```
    Permissions == other.Permissions &&
```

```
    Protocol == other.Protocol &&
```

```
    StartTime == other.StartTime &&
```

```
    Version == other.Version;
```

# Ошибки которые не хочется искать

```
public struct FileSasBuilder : IEquatable<FileSasBuilder> {  
    ....  
    public bool Equals(FileSasBuilder other) =>  
        CacheControl == other.CacheControl  
        && ContentDisposition == other.ContentDisposition  
        && ContentEncoding == other.ContentEncoding  
        && ContentLanguage == other.ContentEncoding  
        && ContentType == other.ContentType  
        && ExpiryTime == other.ExpiryTime  
        && FilePath == other.FilePath  
        && Identifier == other.Identifier  
        && IPRange == other.IPRange  
        && Permissions == other.Permissions  
        && Protocol == other.Protocol  
        && ShareName == other.ShareName  
        && StartTime == other.StartTime  
        && Version == other.Version  
}
```

Azure  
SDK



# Ошибки которые не хочется искать

Azure  
SDK

```
public struct FileSasBuilder : IEquatable<FileSasBuilder> {  
    ....  
    public bool Equals(FileSasBuilder other) =>  
        CacheControl == other.CacheControl  
        && ContentDisposition == other.ContentDisposition  
        && ContentEncoding == other.ContentEncoding  
        && ContentLanguage == other.ContentEncoding  
        && ContentType == other.ContentType  
        && ExpiryTime == other.ExpiryTime  
        && FilePath == other.FilePath  
        && Identifier == other.Identifier  
        && IPRange == other.IPRange  
        && Permissions == other.Permissions  
        && Protocol == other.Protocol  
        && ShareName == other.ShareName  
        && StartTime == other.StartTime  
        && Version == other.Version  
}
```

# Ошибки из-за невнимательности



```
if ( request.PaymentTolerance < 0  
  
    && request.PaymentTolerance > 100)  
{  
    ModelState.AddModelError(nameof(request.PaymentTolerance) ,  
  
    "PaymentTolerance can only be between 0 and 100 percent") ;  
  
    . . .  
}
```

# Ошибки из-за невнимательности



```
if ( request.PaymentTolerance < 0  
  
    &&request.PaymentTolerance > 100)  
{  
    ModelState.AddModelError(nameof(request.PaymentTolerance) ,  
    "PaymentTolerance can only be between 0 and 100 percent") ;  
}
```

## Предупреждение PVS-Studio:

- V3022 Expression 'request.PaymentTolerance < 0 && request.PaymentTolerance > 100' is always false. Probably the '||' operator should be used here.

## Ошибки из-за неочевидных моментов



```
var exp = new Expression(CalcExpression);  
var result = exp.Eval();  
  
if (result == double.NaN)  
{  
    Current = "ERROR";  
    return;  
}
```

## Ошибки из-за неочевидных моментов



```
var exp = new Expression(CalcExpression);  
var result = exp.Eval();
```

```
if (result == double.NaN)  
{
```

### Предупреждение PVS-Studio:

PVS-Studio: V3076 Comparison of 'result' with 'double.NaN' is meaningless. Use 'double.IsNaN()' method instead.

```
}
```



# Entity Framework

Entity Framework



```
for (var i = 0; i < result.Count; i++)
{
    for (var j = 0; j < expectedInnerNames.Count; j++)
    {
        Assert.True(
            result[i].OneToMany_Optional
                .Select(e => e.Name)
                .Contains(expectedInnerNames[i]) );
    }
}
```

# Entity Framework

Entity Framework



```
for (var i = 0; i < result.Count; i++)
{
    for (var j = 0; j < expectedInnerNames.Count; j++)
    {
        Assert.True(
            result[i].OneToMany_Optional
                .Select(e => e.Name)
                .Contains(expectedInnerNames[i]) );
    }
}
```

## eShopOnContainers

```
....  
var firstUrl = new Uri(urlHook, UriKind.Absolute);  
var secondUrl = new Uri(url, UriKind.Absolute);  
  
return firstUrl.Scheme == secondUrl.Scheme &&  
       firstUrl.Port == secondUrl.Port &&  
       firstUrl.Host == firstUrl.Host;
```

## eShopOnContainers

```
....  
var firstUrl = new Uri(urlHook, UriKind.Absolute);  
var secondUrl = new Uri(url, UriKind.Absolute);  
  
return firstUrl.Scheme == secondUrl.Scheme &&  
       firstUrl.Port == secondUrl.Port &&  
       firstUrl.Host == firstUrl.Host;
```

## eShopOnContainers

```
....  
var firstUrl = new Uri(urlHook, UriKind.Absolute);  
var secondUrl = new Uri(url, UriKind.Absolute);  
  
return firstUrl.Scheme == secondUrl.Scheme &&  
       firstUrl.Port == secondUrl.Port &&  
       firstUrl.Host == firstUrl.Host;
```

### Предупреждение PVS-Studio:

V3001 There are identical sub-expressions 'firstUrl.Host' to the left and to the right of the '==' operator.



# eShopOnContainers

```
var parts = path.Split('.');  
for (int i = 0; i < parts.Length; i++)  
{  
    var part = parts[i];  
    if (!(propertyValue is IList))  
        throw new ArgumentException(  
            "Object addressing by pathing segment '{part}'  
            with indexer must be IList");  
  
    if (!(propertyValue is IPropertyBag))  
        throw new ArgumentException(  
            "Object addressing by pathing segment '{part}'  
            must be IPropertyBag");  
    ...  
}
```

# eShopOnContainers

```
var parts = path.Split('.');  
for (int i = 0; i < parts.Length; i++)  
{  
    var part = parts[i];  
    if (!(propertyValue is IList))  
        throw new ArgumentException(  
            "Object addressing by pathing segment 'part'  
            with indexer must be IList");  
  
    if (!(propertyValue is IPropertyBag))  
        throw new ArgumentException(  
            "Object addressing by pathing segment 'part'  
            must be IPropertyBag");  
    ....  
}
```

Могут ли статические  
анализаторы в оптимизацию?

Нет.



Нет.

*Почти :)*





# Оптимизация и статический анализ

- Статический анализ под это **не заточен**



# Оптимизация и статический анализ

- Статический анализ под это **не заточен**
- Есть **МИКРО**оптимизации



# Оптимизация и статический анализ

- Статический анализ под это **не заточен**
- Есть **МИКРО**оптимизации
- Может дополнить профилировщики/динамические анализаторы



# Микрооптимизации

```
inline void setLogTag(const std::string tagName) {  
    m_tag = tagName;  
}
```

# Микрооптимизации

```
inline void setLogTag(const std::string tagName) {  
    m_tag = tagName;  
}
```

## Предупреждение PVS-Studio:

V801 Decreased performance. It is better to redefine the first function argument as a reference.

Consider replacing 'const .. tagName' with 'const .. &tagName'.



# Микрооптимизации

```
inline void setLogTag(const std::string &tagName) {  
    m_tag = tagName;  
}
```



## Предупреждение PVS-Studio:

V801 Decreased performance. It is better to redefine the first function argument as a reference.

Consider replacing 'const .. tagName' with 'const .. &tagName'.

# Микрооптимизации

```
inline void setLogTag(const std::string tagName) {  
    m_tag = std::move(tagName) ;   
}
```

```
void
addDescriptions(std::vector<std::pair<int, std::string>> toAdd)
{
    if (m_descCount + toAdd.size() > MAX_POLICY_DESCRIPTIONS) {
        throw std::length_error("Descriptions count would exceed "
                                + std::to_string(MAX_POLICY_DESCRIPTIONS));
    }
    auto addDesc = [] (DescrType **desc, int result,
                      const std::string &name)
    {
        (*desc) = static_cast<DescrType *>(malloc(sizeof(DdescrType)));
        (*desc)->result = result;
        (*desc)->name = strdup(name.data());
    };
    for (const auto &it : toAdd) {
        addDesc(m_policyDescs + m_descCount, it.first, it.second);
        ++m_descCount;
    }
    m_policyDescs[m_descCount] = nullptr;
}
```

```
void
addDescriptions(std::vector<std::pair<int, std::string>> toAdd)
{
    if (m_descCount + toAdd.size() > MAX_POLICY_DESCRIPTIONS) {
        throw std::length_error("Descriptions count would exceed "
                                + std::to_string(MAX_POLICY_DESCRIPTIONS));
    }
    auto addDesc = [] (DescrType **desc, int result,
                      const std::string &name)
    {
        (*desc) = static_cast<DescrType *>(malloc(sizeof(DdescrType)));
        (*desc)->result = result;
        (*desc)->name = strdup(name.data());
    };
    for (const auto &it : toAdd) {
        addDesc(m_policyDescs + m_descCount, it.first, it.second);
        ++m_descCount;
    }
    m_policyDescs[m_descCount] = nullptr;
}
```

```
void  
addDescriptions (std::vector<std::pair<int, std::string>> toAdd)  
{  
    if (m_descCount + toAdd.size() > MAX_POLICY_DESCRIPTIONS)  
        {.....};  
    for (const auto &it : toAdd)  
        {.....};  
}
```




```
void  
addDescriptions (std::vector<std::pair<int, std::string>> toAdd)  
{  
    if (m_descCount + toAdd.size() > MAX_POLICY_DESCRIPTIONS)  
        {.....};  
    for (const auto &it : toAdd)  
        {.....};  
}
```

```
void  
addDescriptions (std::vector<std::pair<int, std::string>> toAdd)  
{  
    if (m_descCount + toAdd.size() > MAX_POLICY_DESCRIPTIONS)  
        {.....};  
    for (const auto &it : toAdd)  
        {.....};  
}
```

### Предупреждение PVS-Studio:

V813 Decreased performance. The 'toAdd' argument should probably be rendered as a constant reference.

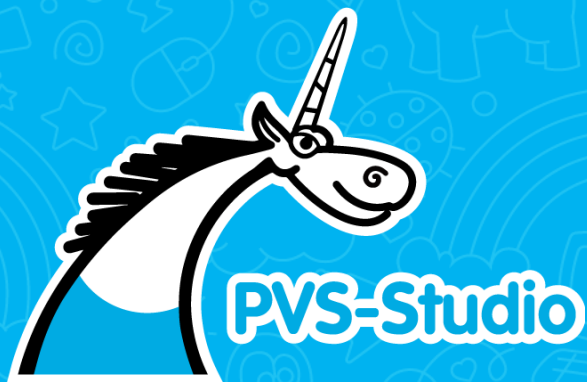
```
void  
addDescriptions (std::vector<std::pair<int, std::string>> &toAdd)  
{  
    if (m_descCount + toAdd.size() > MAX_POLICY_DESCRIPTION)  
        {....};  
    for (const auto &it : toAdd)  
        {....};  
}
```



### Предупреждение PVS-Studio:

V813 Decreased performance. The 'toAdd' argument should probably be rendered as a constant reference.

# SAST: безопасность и защищенность



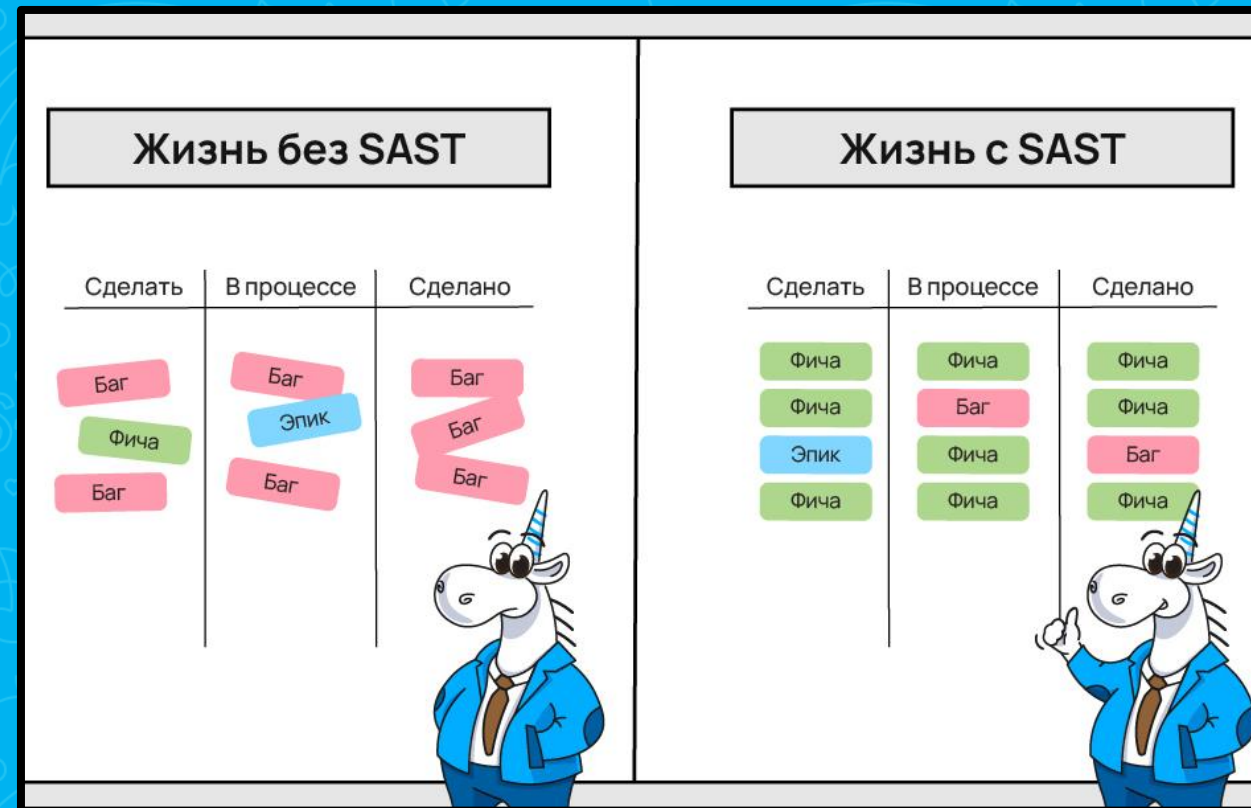






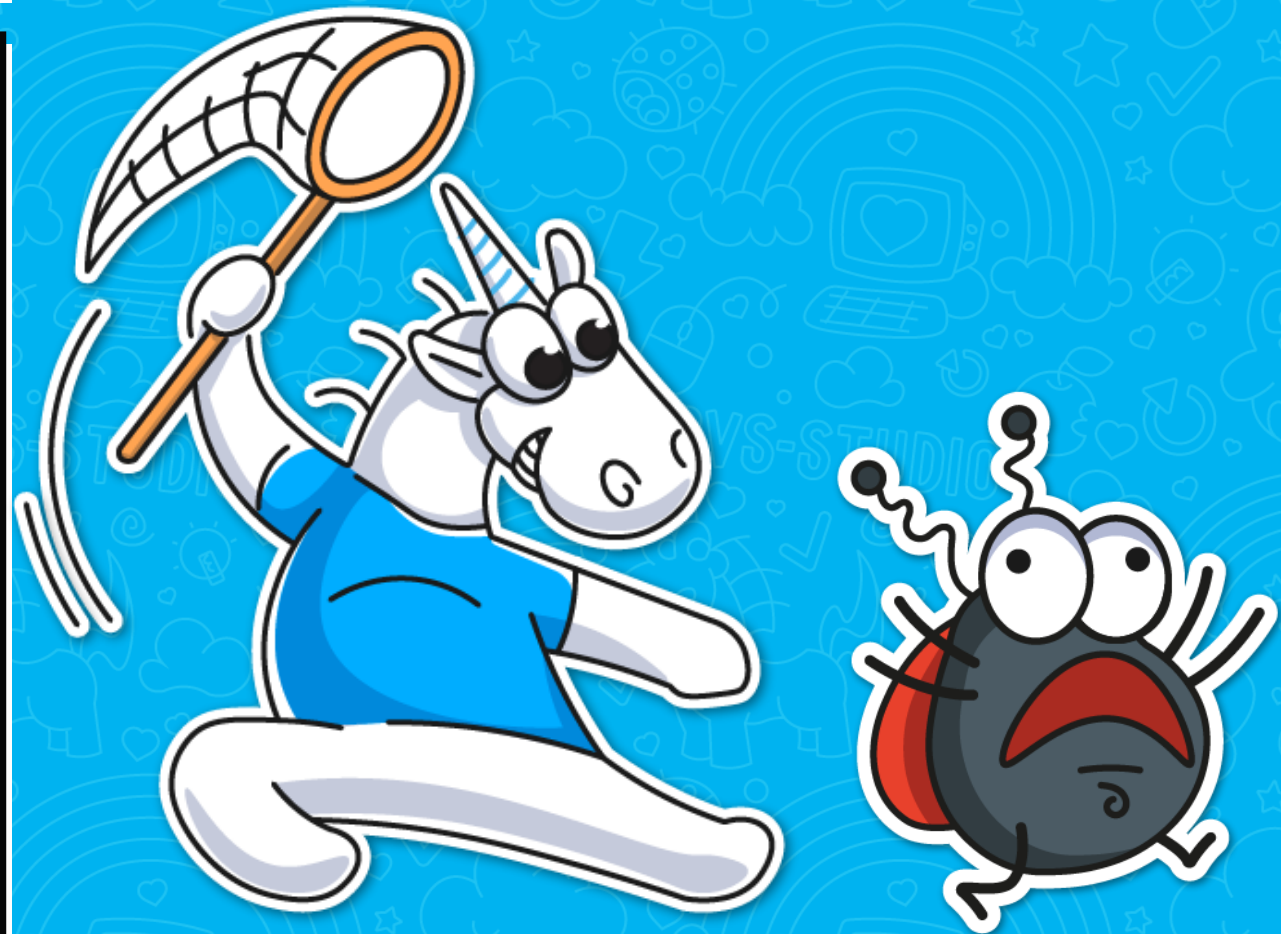
# Что такое SAST?

- Static Application Security Testing
- Статический анализ, но про уязвимости
- Уязвимости — такие-же обычные ошибки



## Безопасность и защищённость

- Safety (безопасность) /  
Security (защищённость)



## Безопасность и защищённость

- **Safety** (безопасность) / Security (защищённость)

### Безопасность

- Надёжная работа приложения в любых условиях, без вмешательства извне
- MISRA C/C++
- AUTOSAR C++



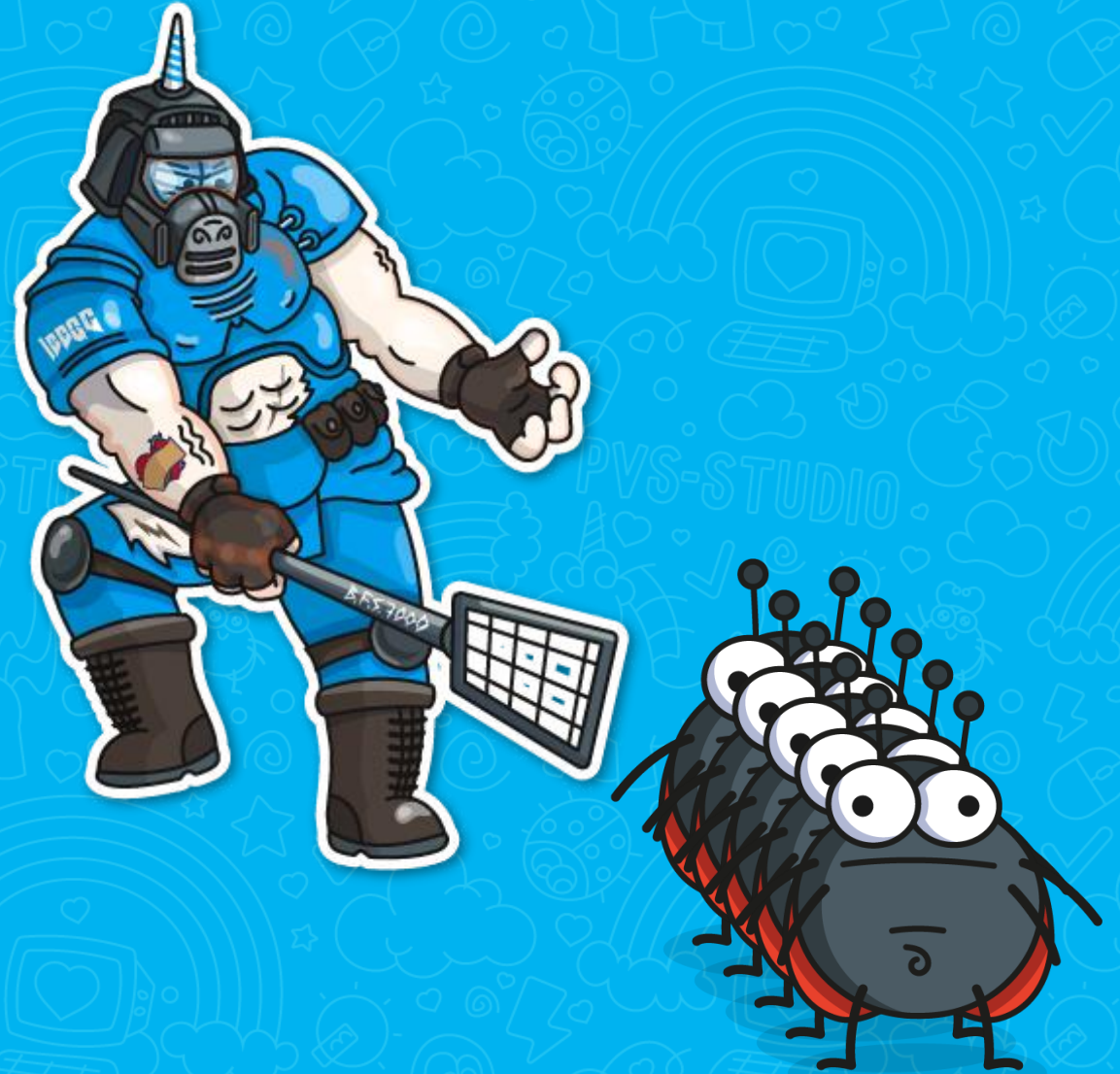


# Безопасность и защищённость

- Safety (безопасность) / **Security** (защищённость)

## Защищённость

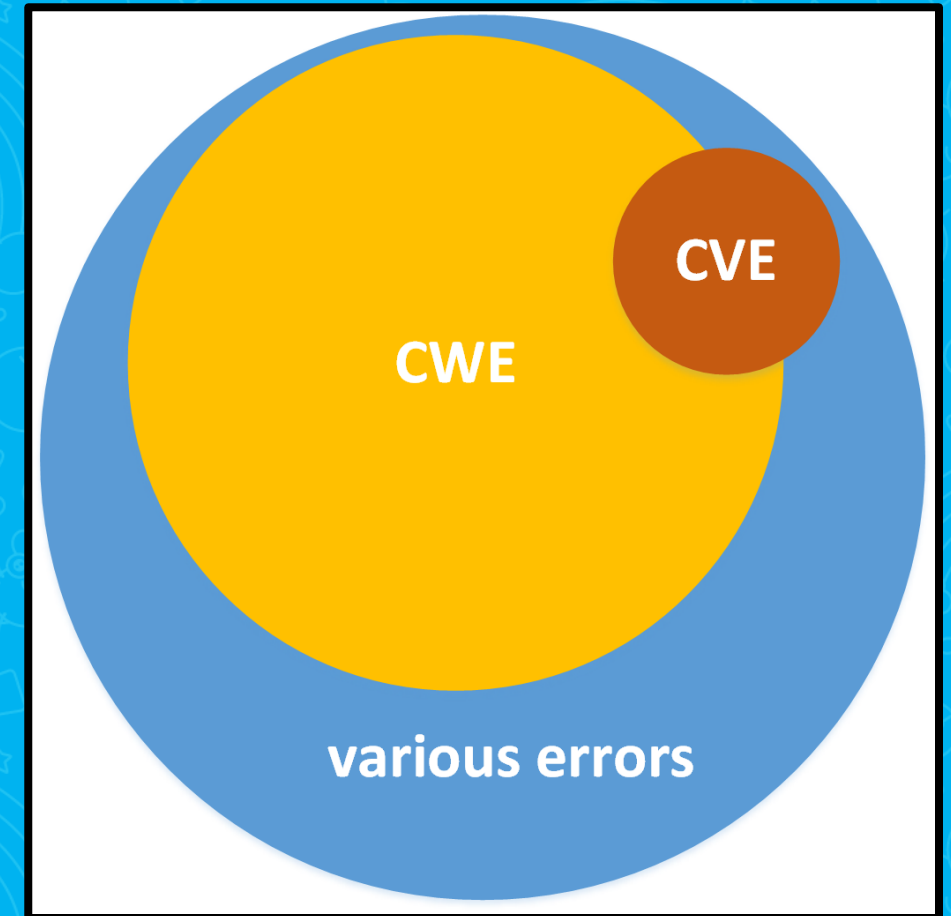
- Стойкость ко внешним воздействиям, попыткам вмешательства извне
- OWASP ASVS
- SEI CERT



## Списки уязвимостей

**CWE** – Потенциальные  
уязвимости  
Common Weakness Enumeration

**CVE** - Существующие  
уязвимости  
Common Vulnerabilities and  
Exposures





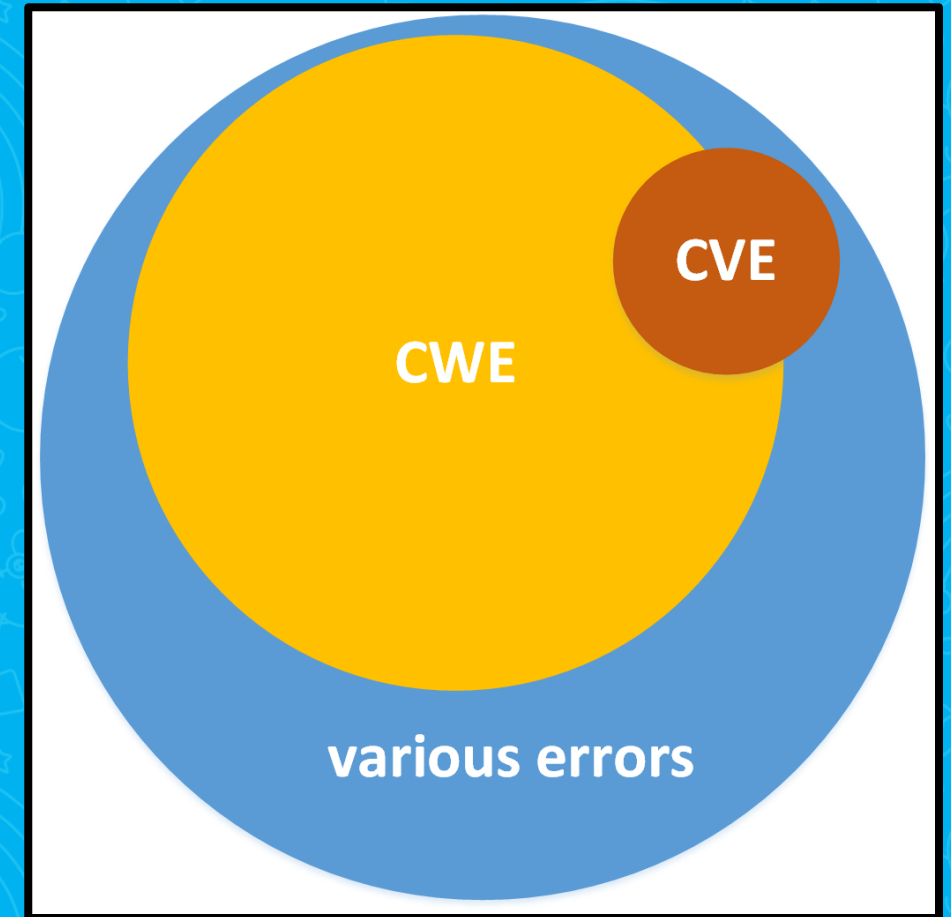
# Списки уязвимостей

**CWE** – Потенциальные  
**уязвимости**

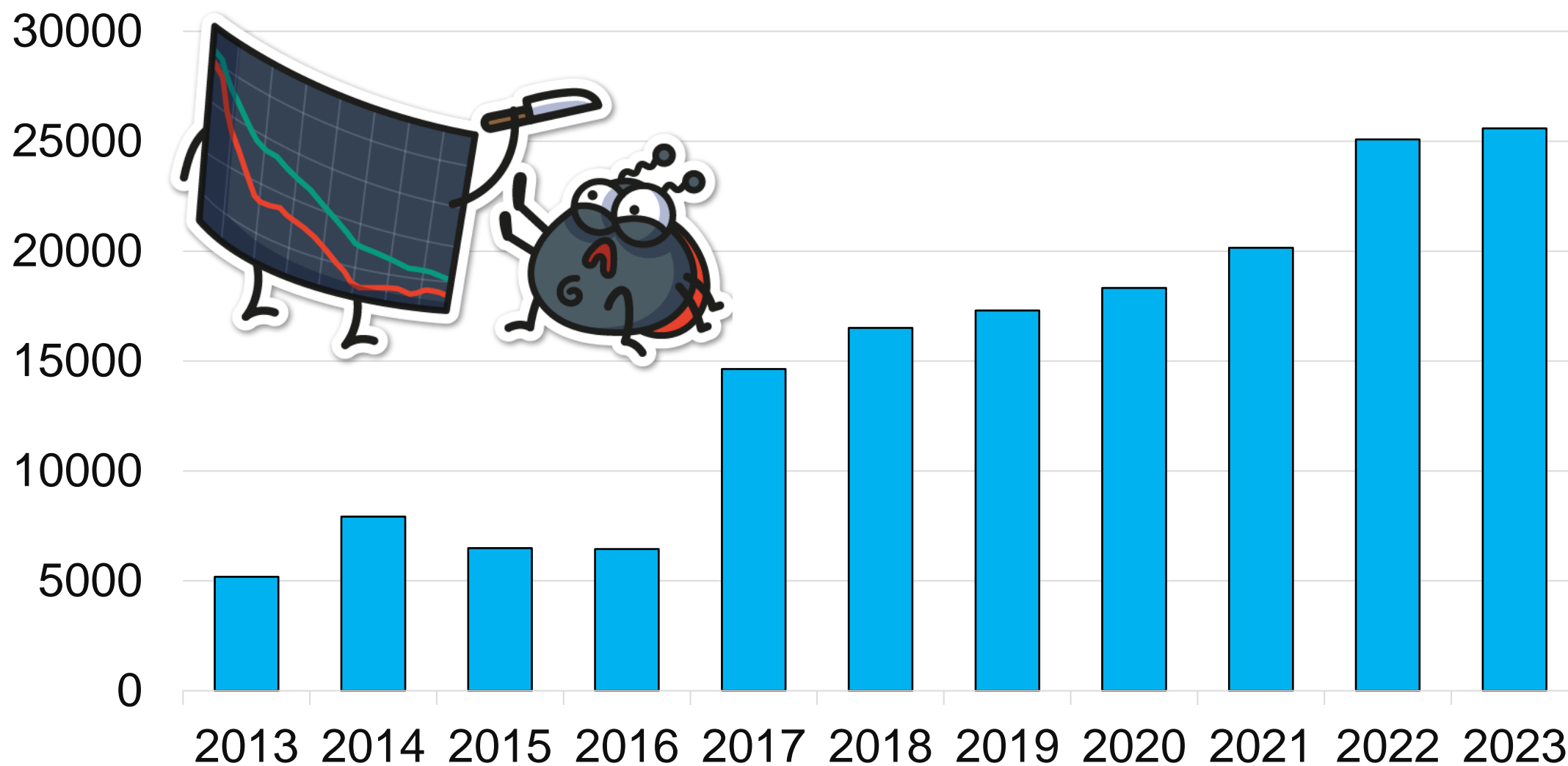
Common Weakness Enumeration

**CVE** - Существующие  
**уязвимости**

Common Vulnerabilities and  
Exposures



# Число выявленных уязвимостей



Источник: [cvedetails.com](https://cvedetails.com)

# SDLC



## Пример CWE в проекте FastReport

```
ParagraphFormat paragraphFormat;
```

```
....
```

```
public ParagraphFormat ParagraphFormat  
{  
    get { return paragraphFormat; }  
    set { ParagraphFormat = value; }  
}
```




## Пример CWE в проекте FastReport

ParagraphFormat paragraphFormat; Поле 

```
....  
public ParagraphFormat ParagraphFormat  
{  
    get { return paragraphFormat; }  
    set { ParagraphFormat = value; }  
}
```

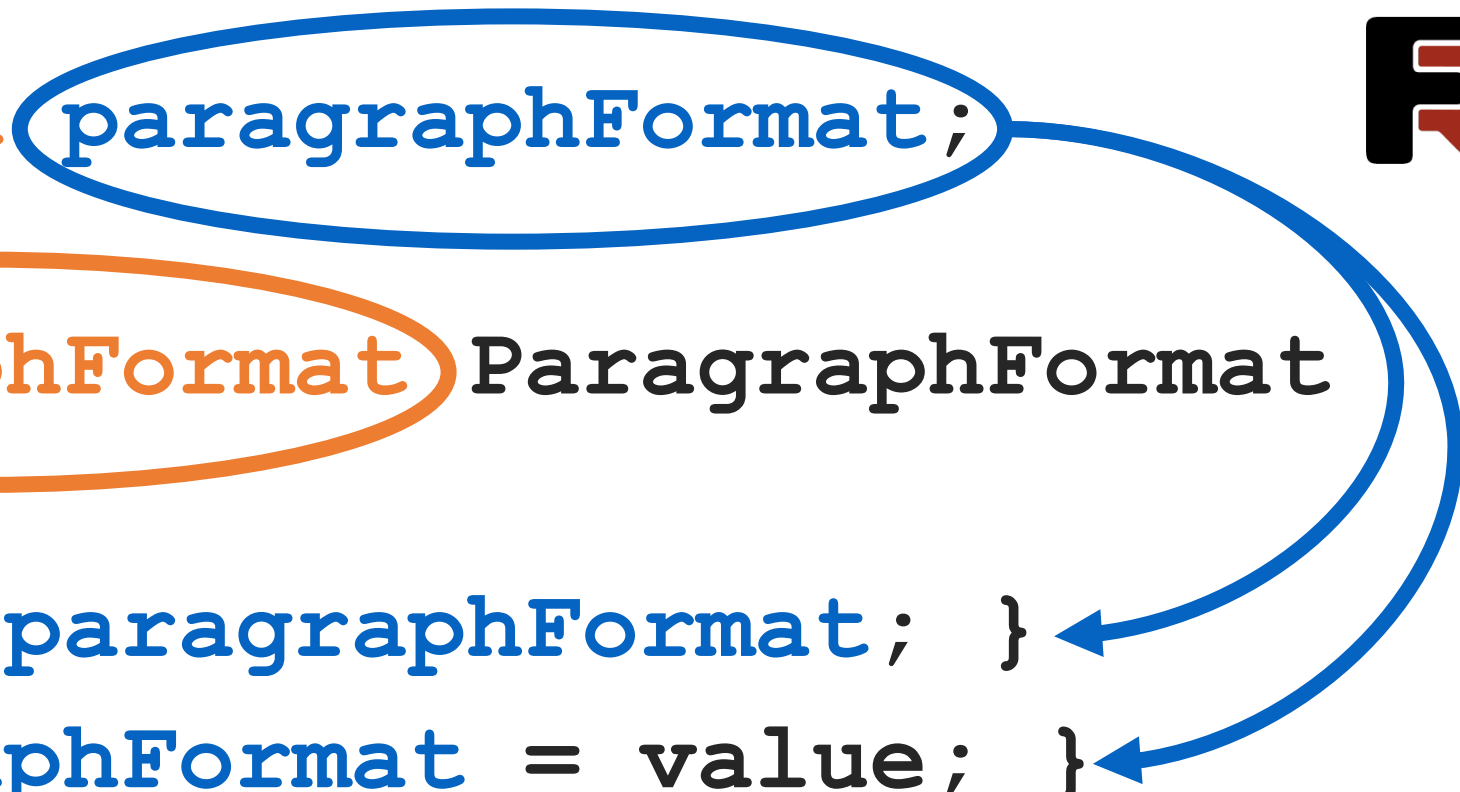
## Пример CWE в проекте FastReport

```
ParagraphFormat paragraphFormat; Поле   
.....  
public ParagraphFormat ParagraphFormat  
{  
    СВОЙСТВО  
    get { return paragraphFormat; }  
    set { ParagraphFormat = value; }  
}
```



## Пример CWE в проекте FastReport

```
ParagraphFormat paragraphFormat;  
....  
public ParagraphFormat ParagraphFormat  
{  
    get { return paragraphFormat; }  
    set { ParagraphFormat = value; }  
}
```

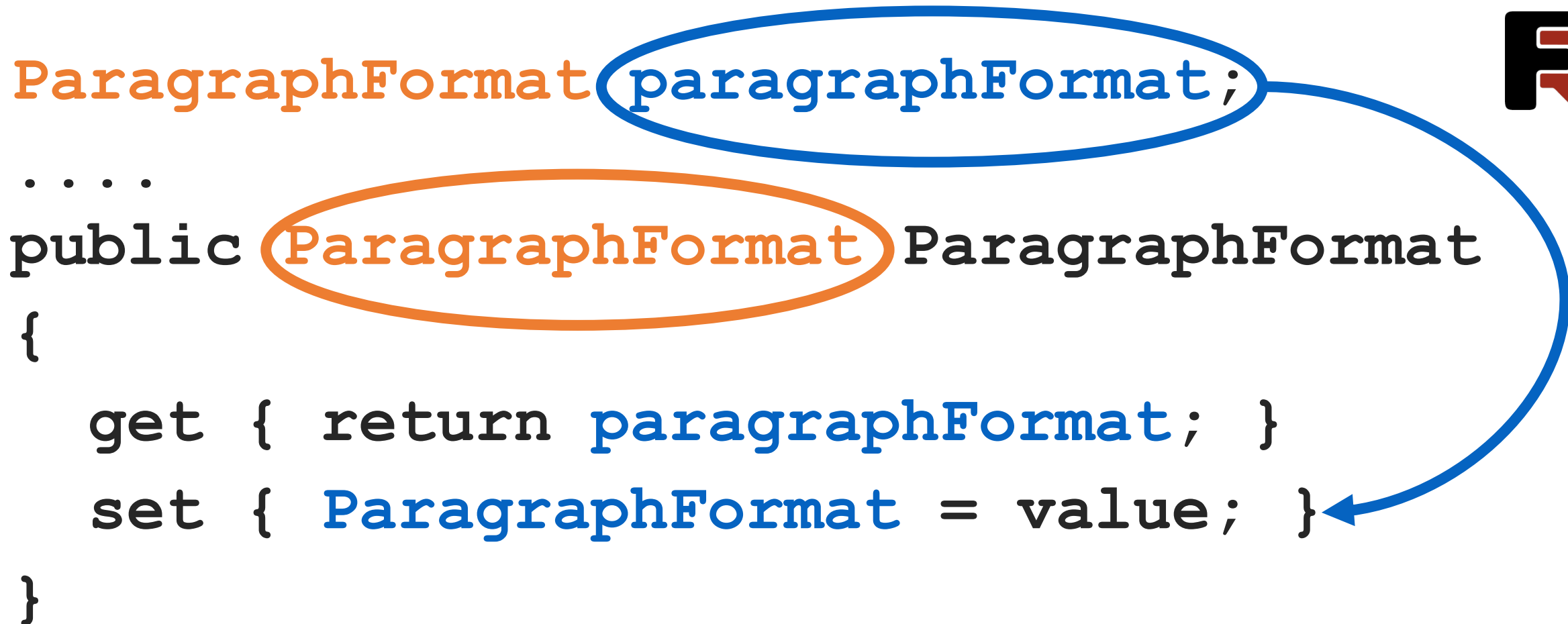


The diagram illustrates a conflict in the FastReport project. A variable `paragraphFormat` (circled in blue) is declared at the top. Below it, a class `ParagraphFormat` (circled in orange) is defined. The class has a `get` method that returns `paragraphFormat` and a `set` method that assigns `ParagraphFormat` to `value`. Blue arrows point from the variable `paragraphFormat` to the `get` and `set` methods, highlighting the conflict where the same name is used for both a variable and a class.



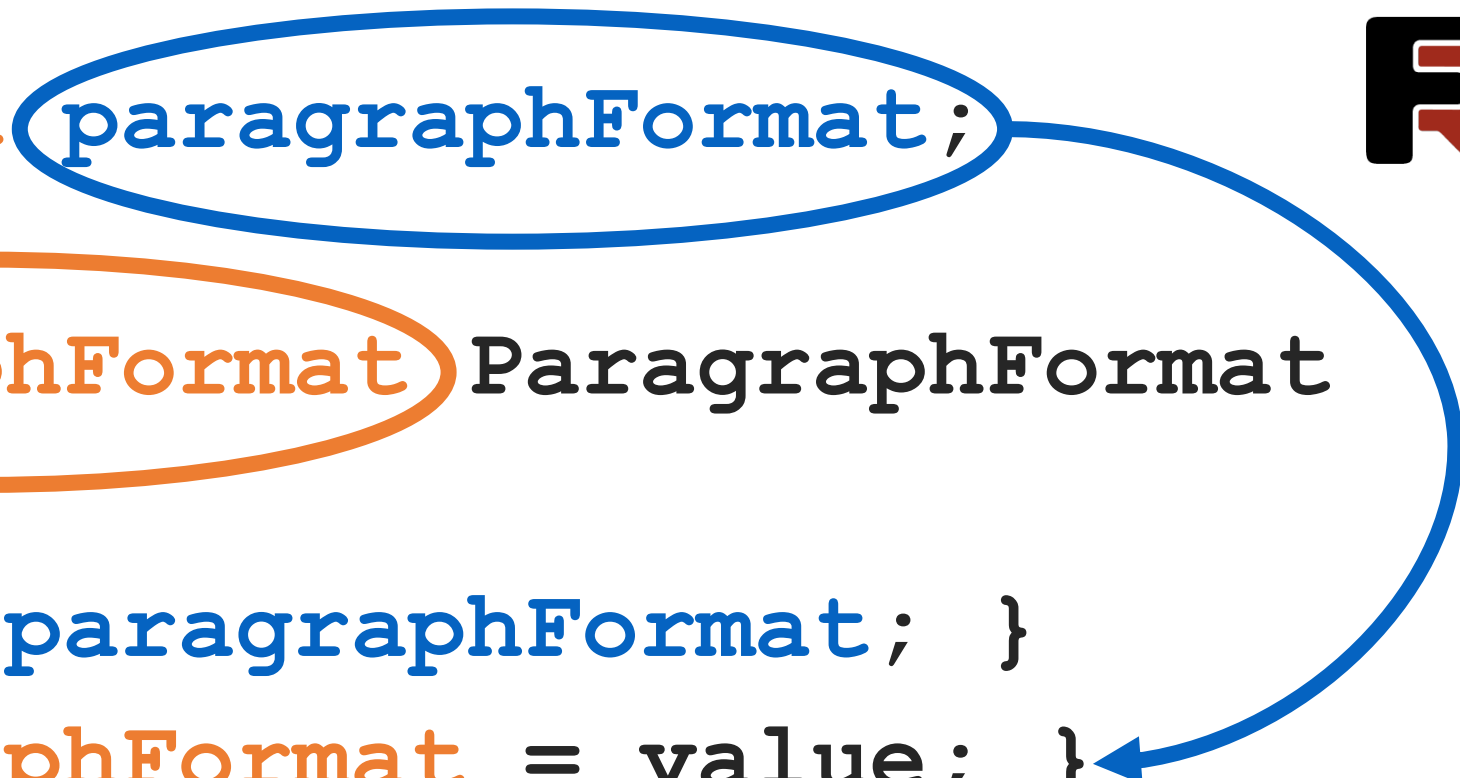
## Пример CWE в проекте FastReport

```
ParagraphFormat paragraphFormat;  
....  
public ParagraphFormat ParagraphFormat  
{  
    get { return paragraphFormat; }  
    set { ParagraphFormat = value; }  
}
```



## Пример CWE в проекте FastReport

```
ParagraphFormat paragraphFormat;  
....  
public ParagraphFormat ParagraphFormat  
{  
    get { return paragraphFormat; }  
    set { ParagraphFormat = value; }  
}
```



The diagram illustrates a conflict in the FastReport project. A blue oval highlights the variable `paragraphFormat` in the first line of code. Another orange oval highlights the class name `ParagraphFormat` in the class definition. A blue arrow originates from the variable `paragraphFormat` and points to the `ParagraphFormat` parameter in the `set` method, indicating that the variable is being used to pass an instance of the class to its own setter, which is a common source of confusion or error.



## Пример CWE в проекте FastReport

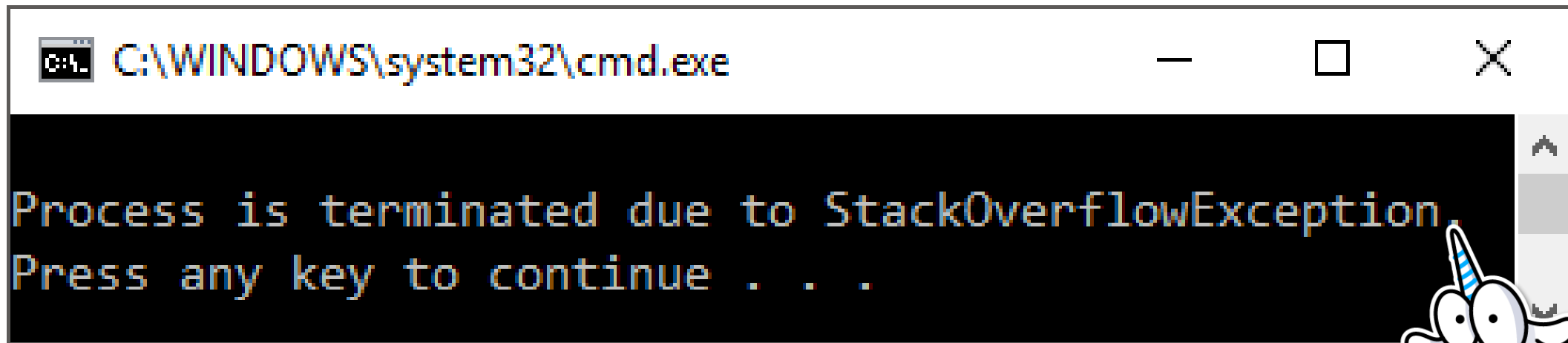
```
static void Main(string[] args)
{
    TextObject textObj = new TextObject();
    textObj.ParagraphFormat = null;

    Console.WriteLine("Ok");
}
```



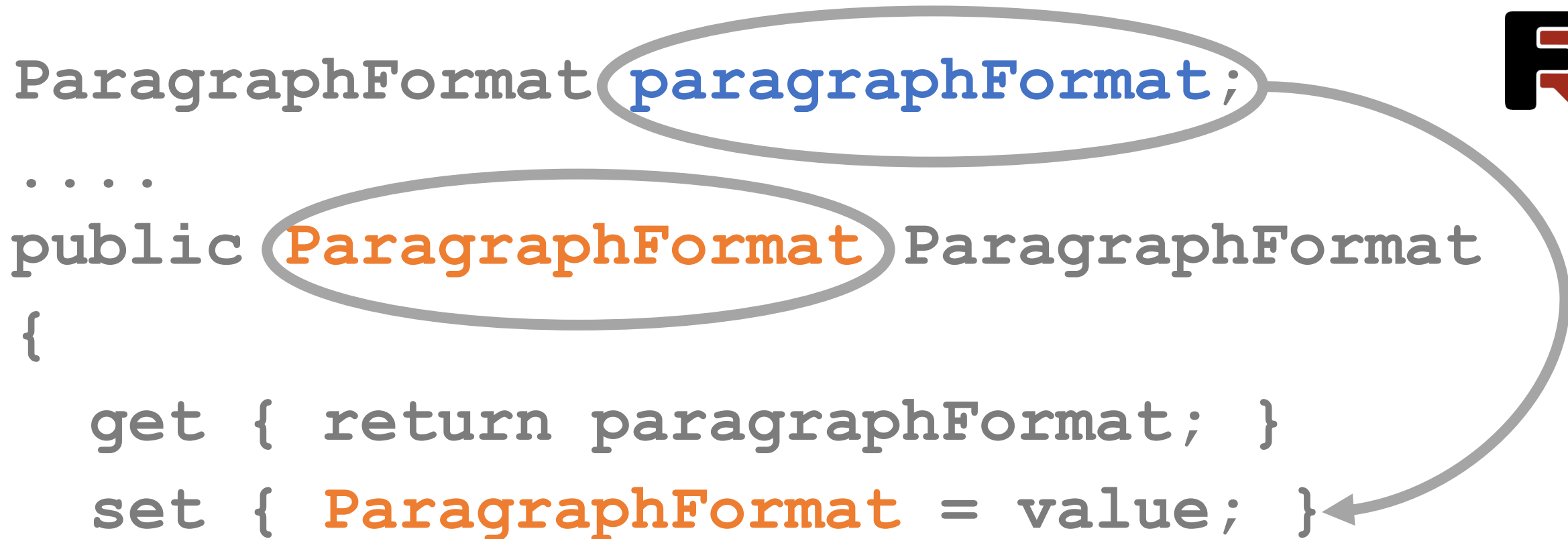
## Пример CWE в проекте FastReport

```
static void Main(string[] args)
{
    Console.WriteLine("Ok");
}
```



## Пример CWE в проекте FastReport

```
ParagraphFormat paragraphFormat;  
....  
public ParagraphFormat ParagraphFormat  
{  
    get { return paragraphFormat; }  
    set { ParagraphFormat = value; }
```



Предупреждение PVS-Studio: V3010 [[CWE-674](#)]

Possible infinite recursion inside 'ParagraphFormat' property.



# Пример уязвимости SQL injection

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

# Пример уязвимости SQL injection

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

# Пример уязвимости SQL injection

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

## Пример уязвимости SQL injection

```
using (SqlConnection connection = new SqlConnection(...))  
{
```

```
....
```

```
SELECT * FROM Users WHERE UserName = 'Иван'
```

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

```
....
```

```
}
```

# Пример уязвимости SQL injection

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



# Пример уязвимости SQL injection

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

# Пример уязвимости SQL injection

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

# Пример уязвимости SQL injection

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

Иван



# Пример уязвимости SQL injection

```
using (SqlConnection connection = new SqlConnection(...))  
{  
    ....  
    String userName =  
using (var command = new SqlCommand("SELECT * FROM Users WHERE UserName = '" + userName + "'", connection))  
    {  
        CommandText =  
            "SELECT * FROM Users WHERE UserName = '" + userName + "'  
        CommandType = System.Data.CommandType.Text }  
    }  
    ....  
}
```

' OR '1'='1'



## Пример уязвимости SQL injection

```
using (SqlConnection connection = new SqlConnection(...))  
{
```

```
    ....
```

```
    SELECT * FROM Users WHERE UserName = 'Иван'
```

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

```
    ....
```

```
}
```



# Пример уязвимости SQL injection

```
using (SqlConnection connection = new SqlConnection(...))  
{
```

```
....
```

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

```
SELECT * FROM Users WHERE UserName = ' ' OR '1'='1'
```

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

```
....
```

```
}
```



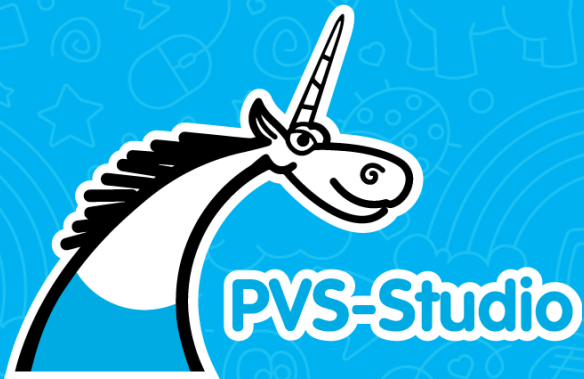
# Пример уязвимости SQL injection

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


Предупреждение PVS-Studio:

**V5608** Possible SQL injection. Potentially tainted data in the 'userName' variable is used to create SQL command.

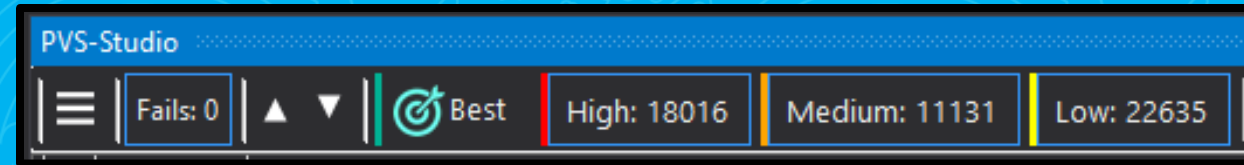
## Особенности при интеграции в legacy проект



PVS-Studio

≡ Fails: 0 ▲ ▼  Best High: 18016 Medium: 11131 Low: 22635

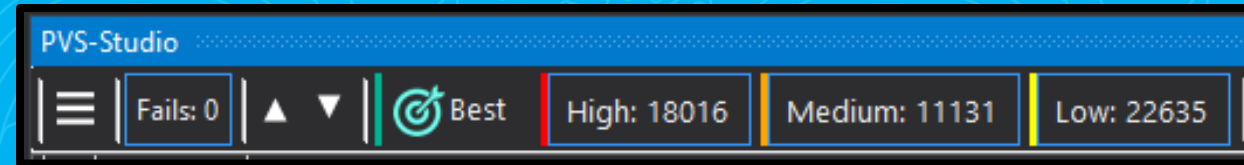
# Опасность первого раза





# Опасность первого раза

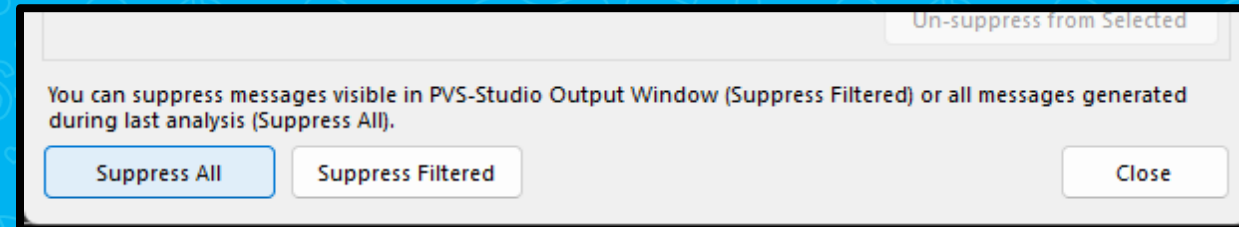
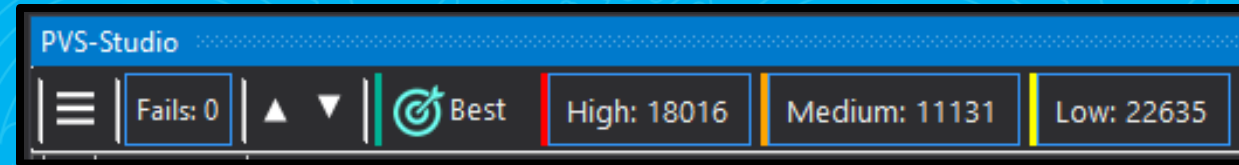
- Много срабатываний - нормально





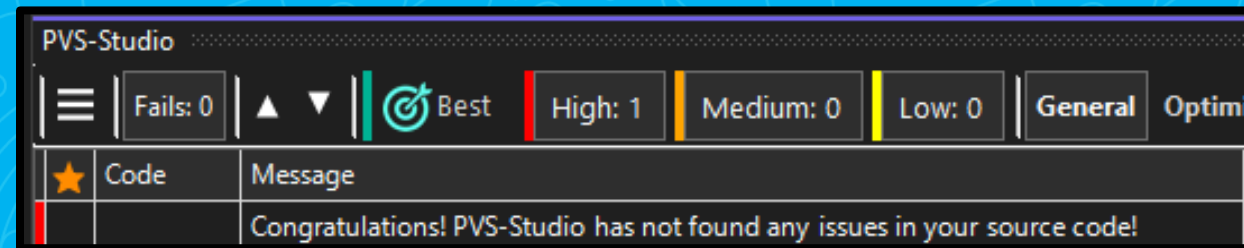
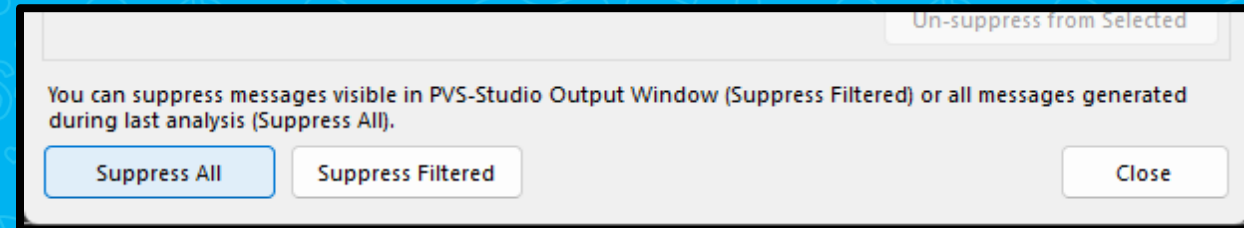
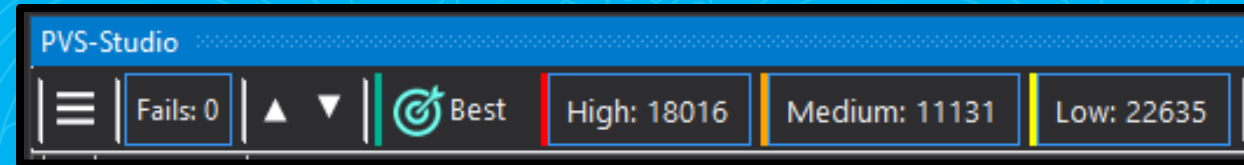
# Опасность первого раза

- Много срабатываний - нормально
- **Используем массовое подавление**



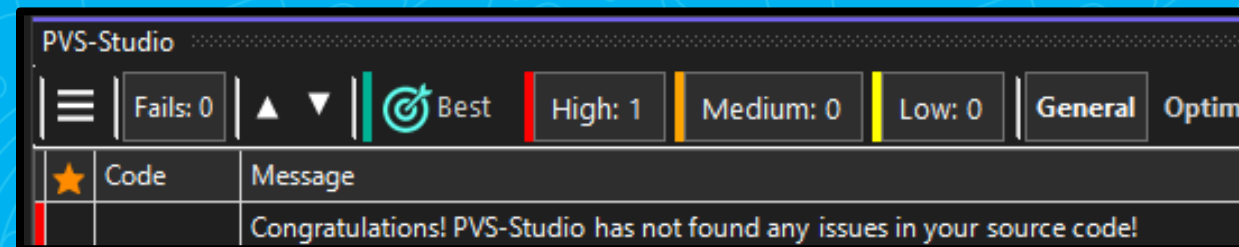
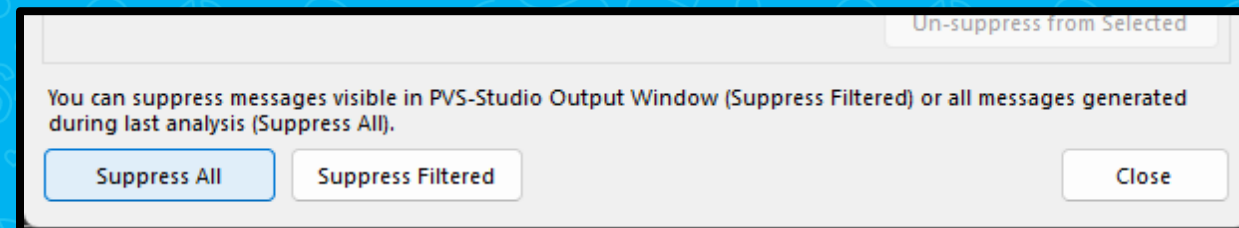
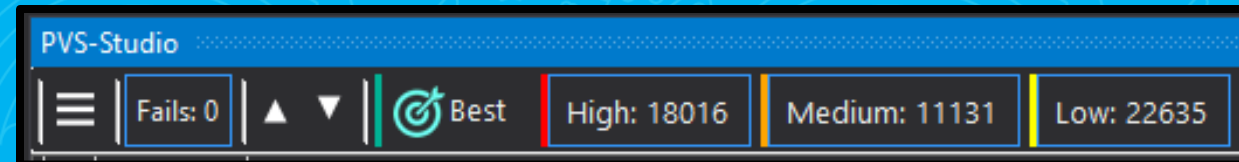
# Опасность первого раза

- Много срабатываний - нормально
- **Используем массовое подавление**
- Периодически возвращаемся и чиним

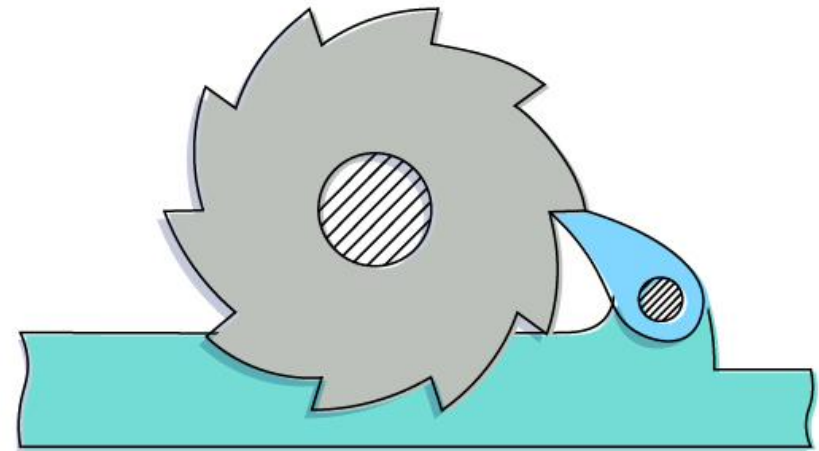


# Опасность первого раза

- Много срабатываний - нормально
- **Используем массовое подавление**
- Периодически возвращаемся и чиним
- Но есть и другой способ...

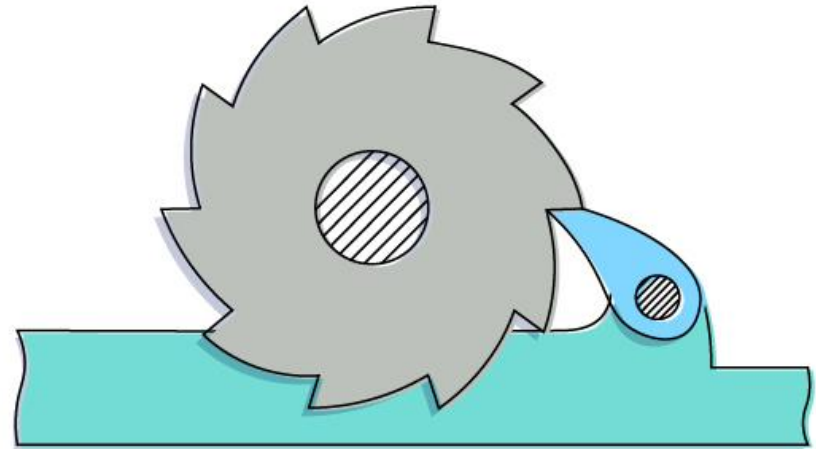


# Принцип Храповика



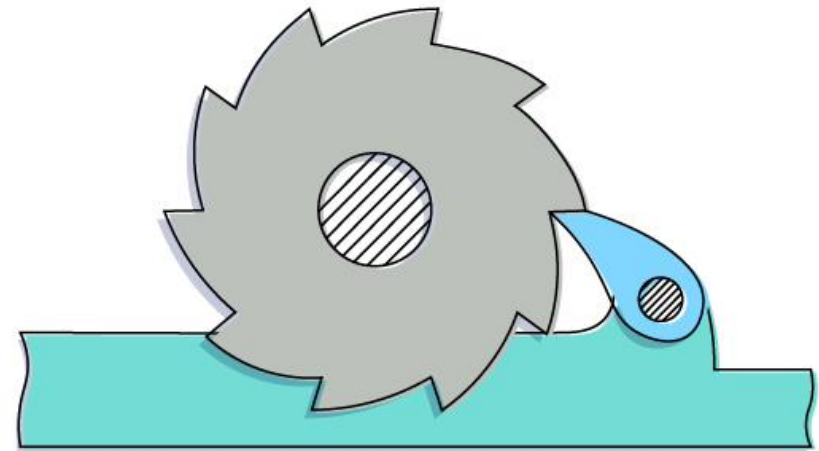
# Принцип Храповика

- Выполняем анализ



# Принцип Храповика

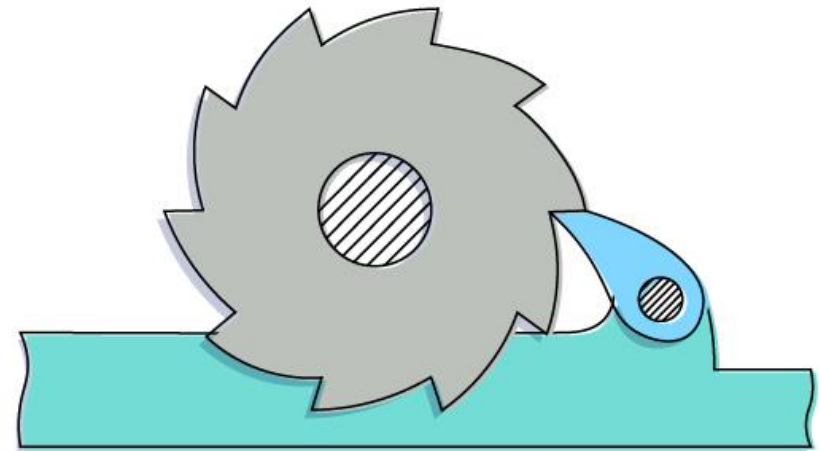
- Выполняем анализ
- **Заносим в систему контроля версий и устанавливаем порог вхождения**



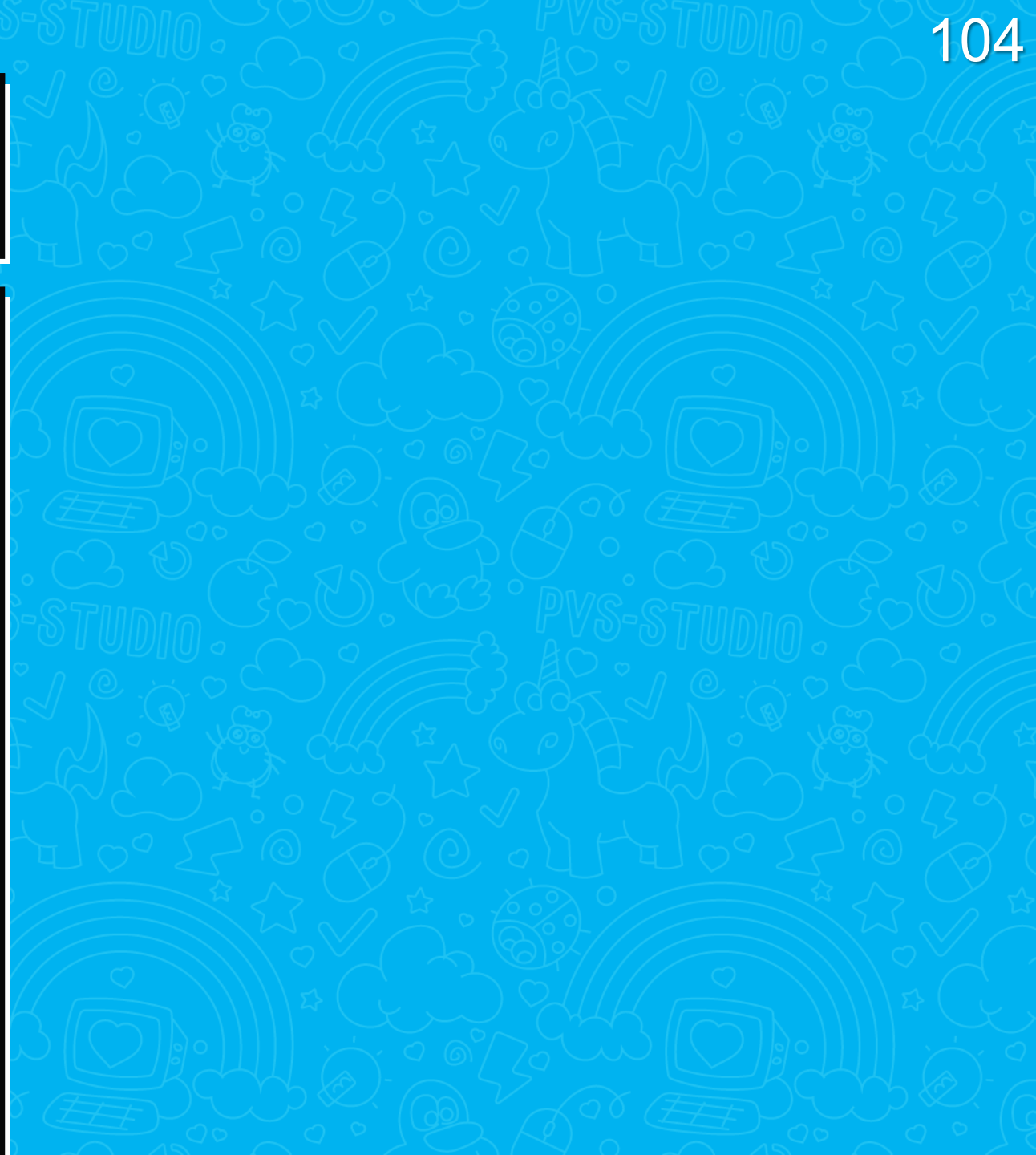


# Принцип Храповика

- Выполняем анализ
- **Заносим в систему контроля версий и устанавливаем порог вхождения**
- Исправляем!



# Ложные срабатывания

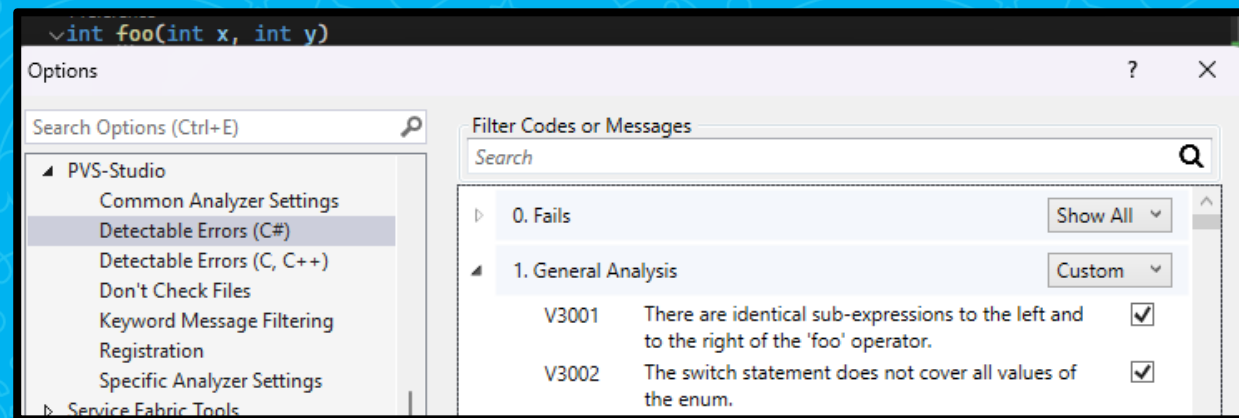


# Ложные срабатывания

- Особенность технологии

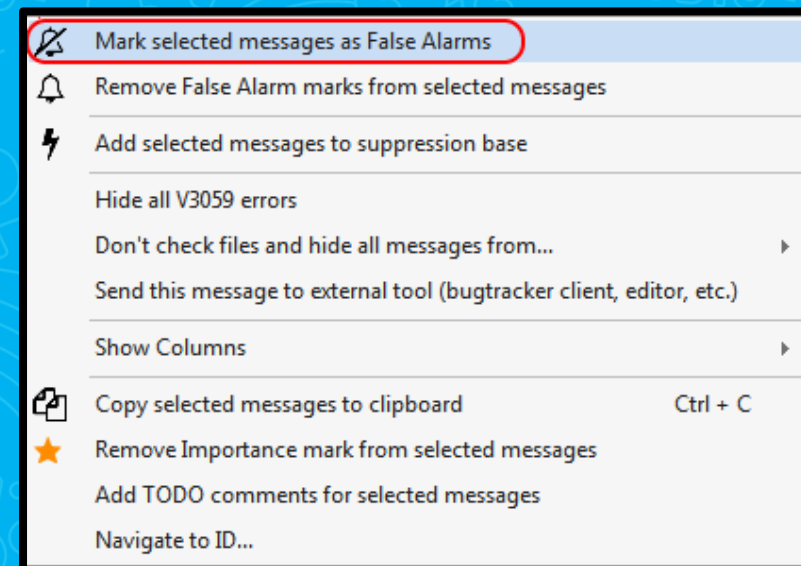
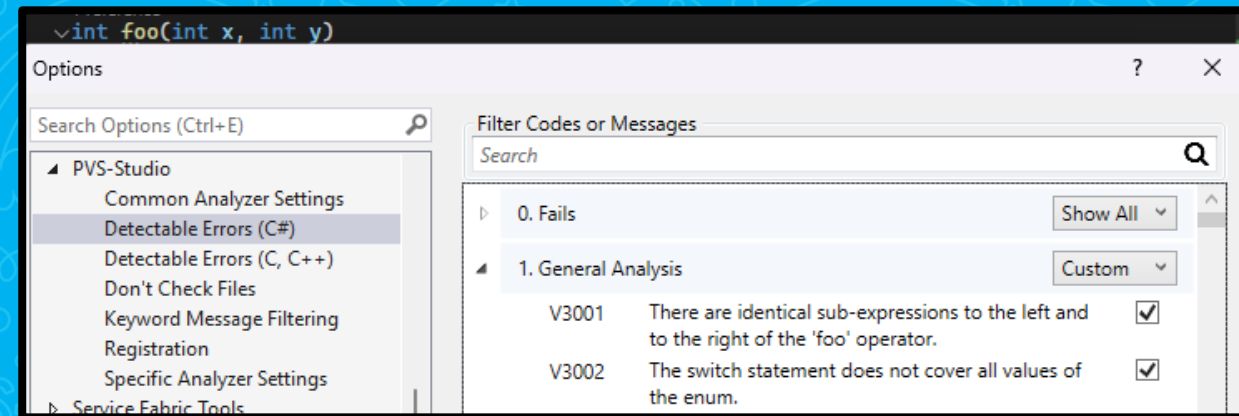
# Ложные срабатывания

- Особенность технологии
- Настраиваем анализатор под проект



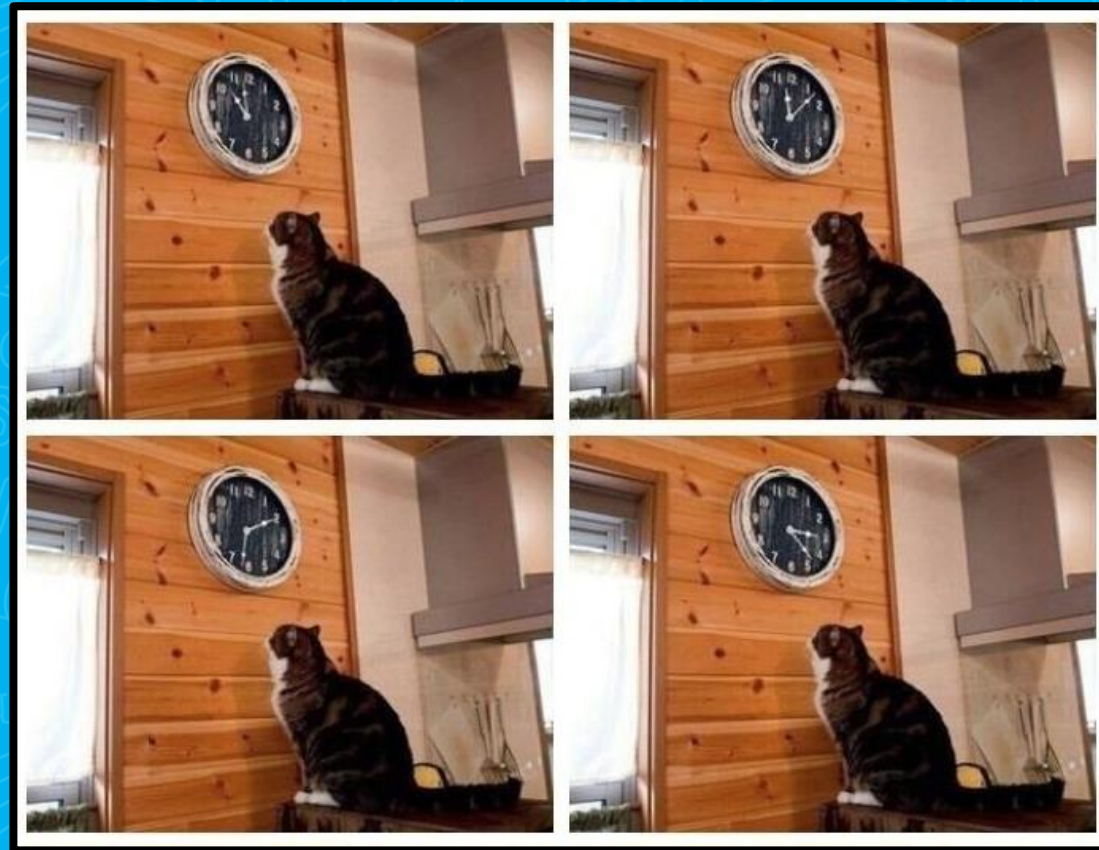
# Ложные срабатывания

- Особенность технологии
- Настраиваем анализатор под проект
- Полностью избавиться не получится, но можно уменьшить вероятность ложного срабатывания





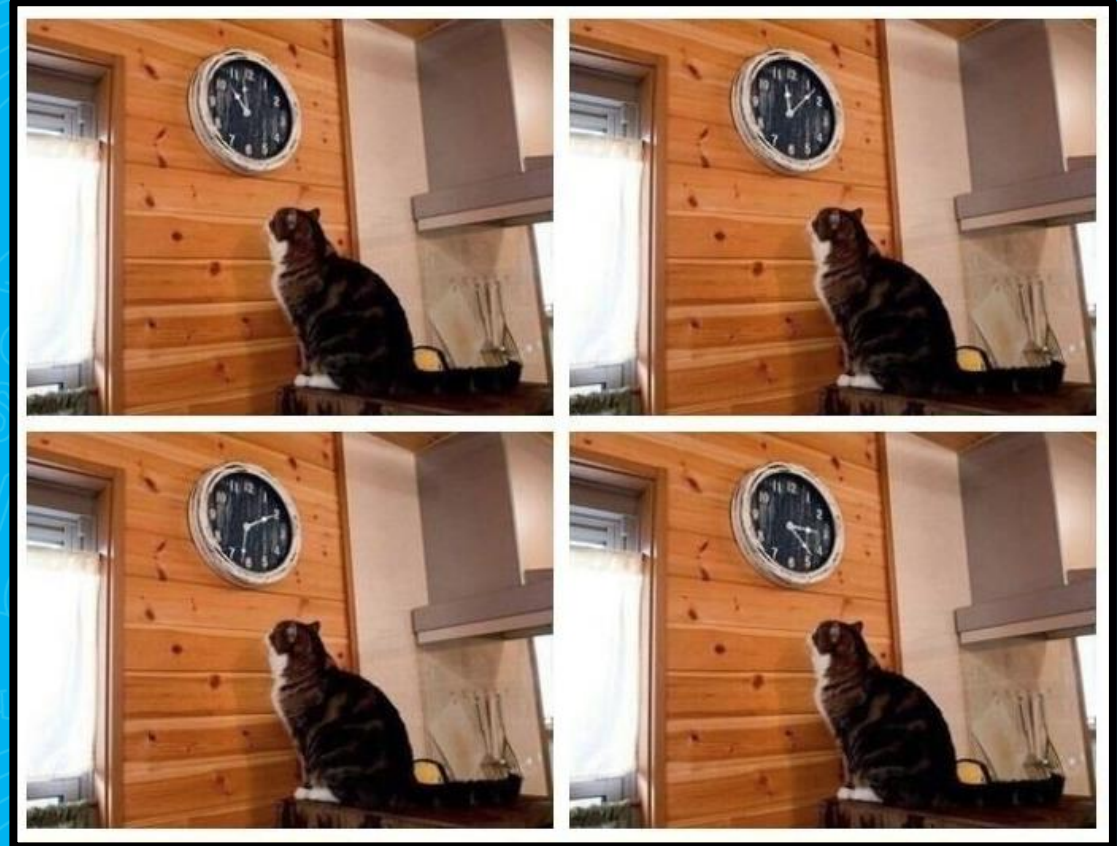
# Долго время анализа





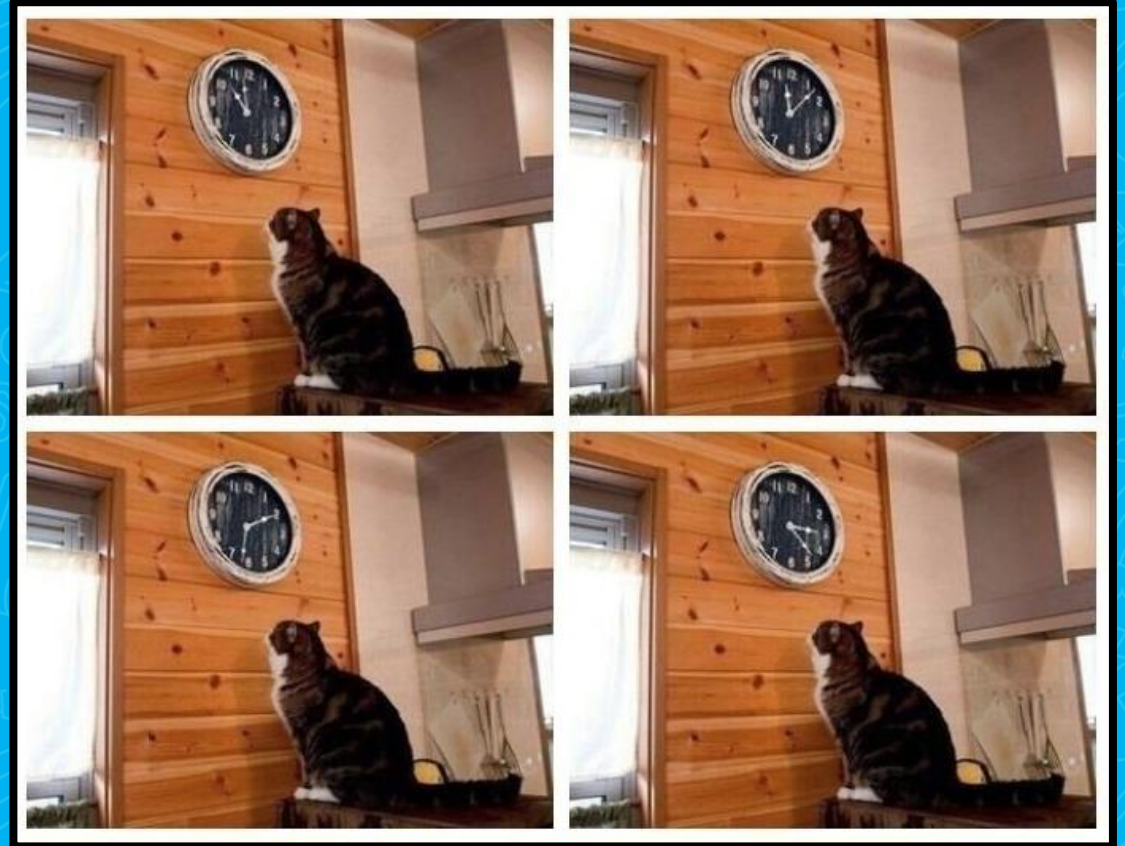
# Долго время анализа

- Больше проект = больше время



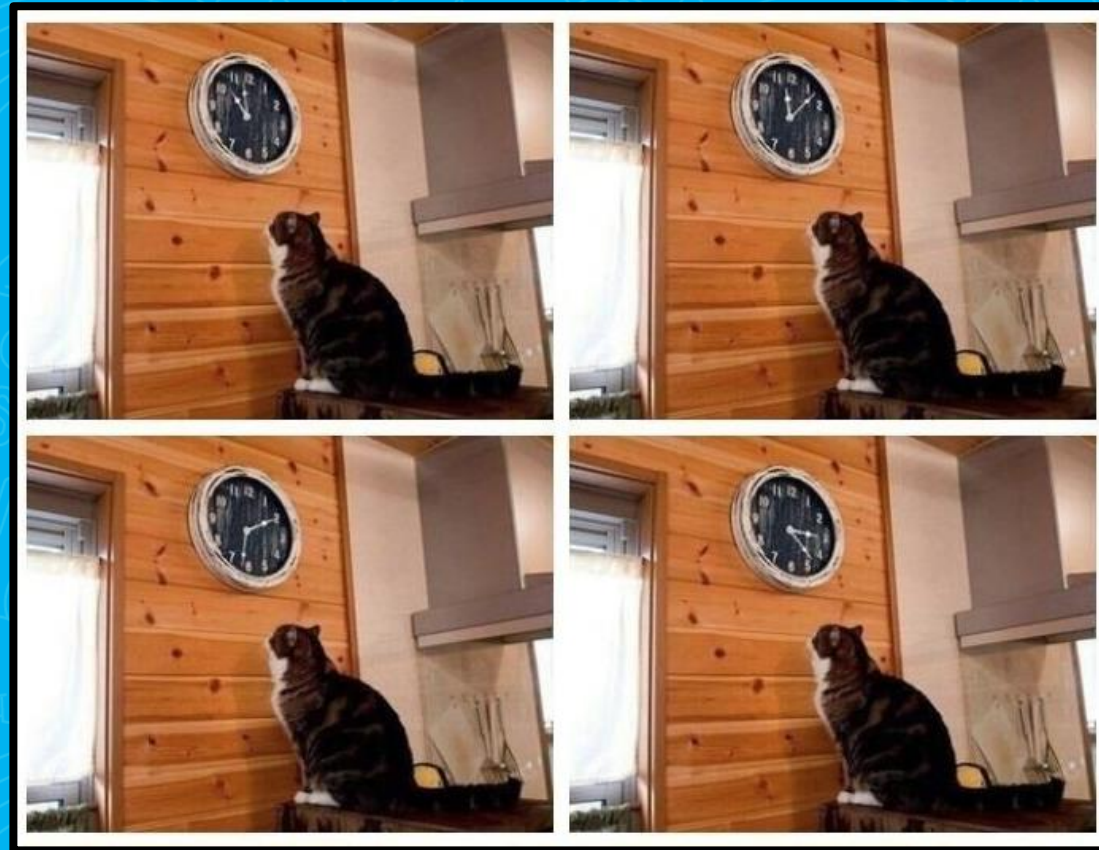
# Долго время анализа

- Больше проект = больше время
- **Инкрементальный анализ**



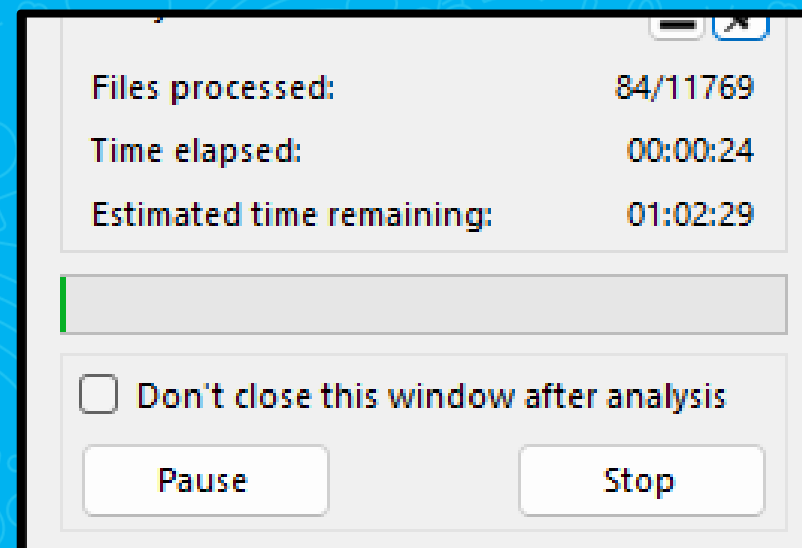
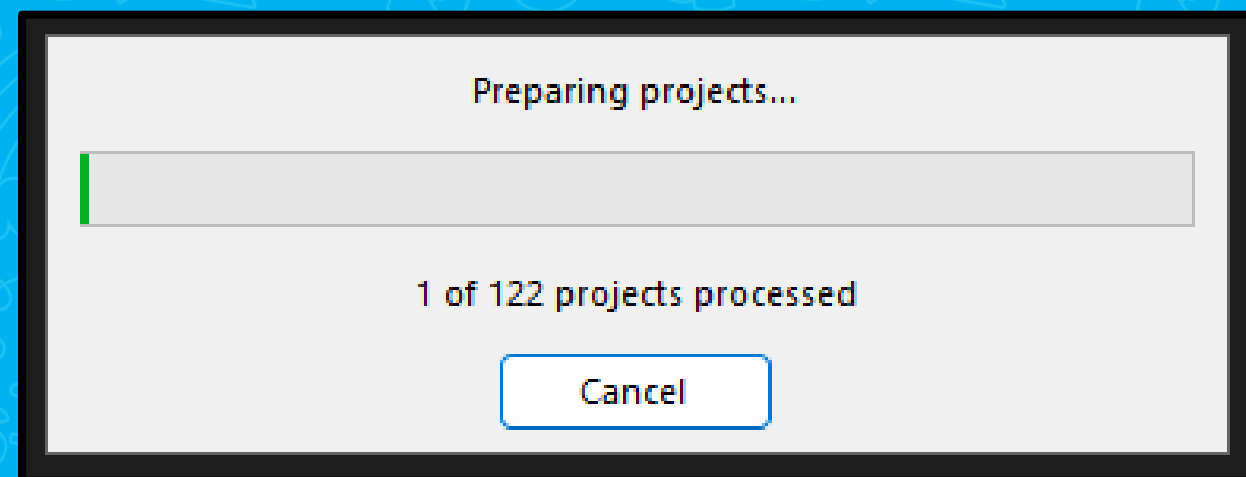
# Долго время анализа

- Больше проект = больше время
- **Инкрементальный анализ**
- Проверяйте только **изменённый код**



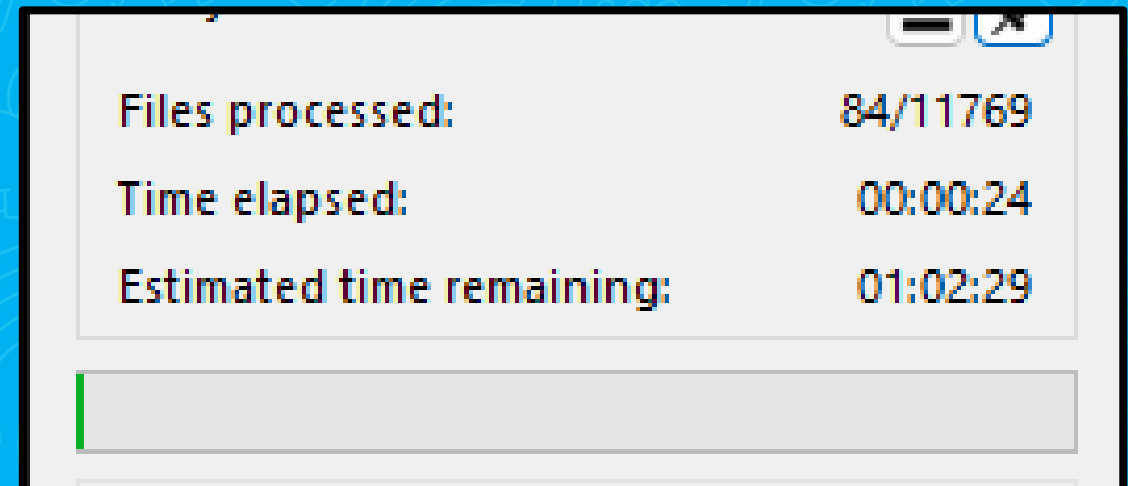
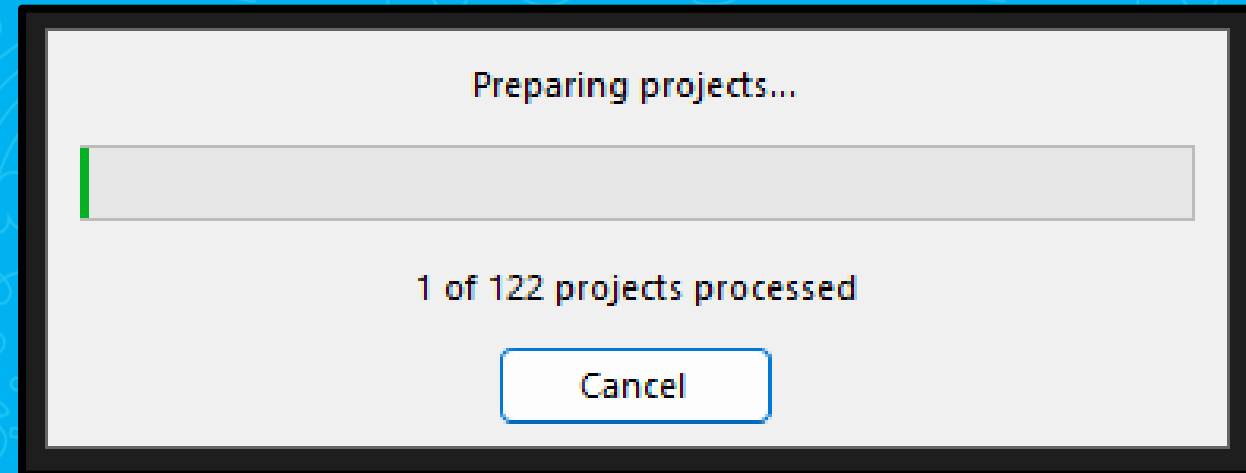


## «Избыточный» анализ



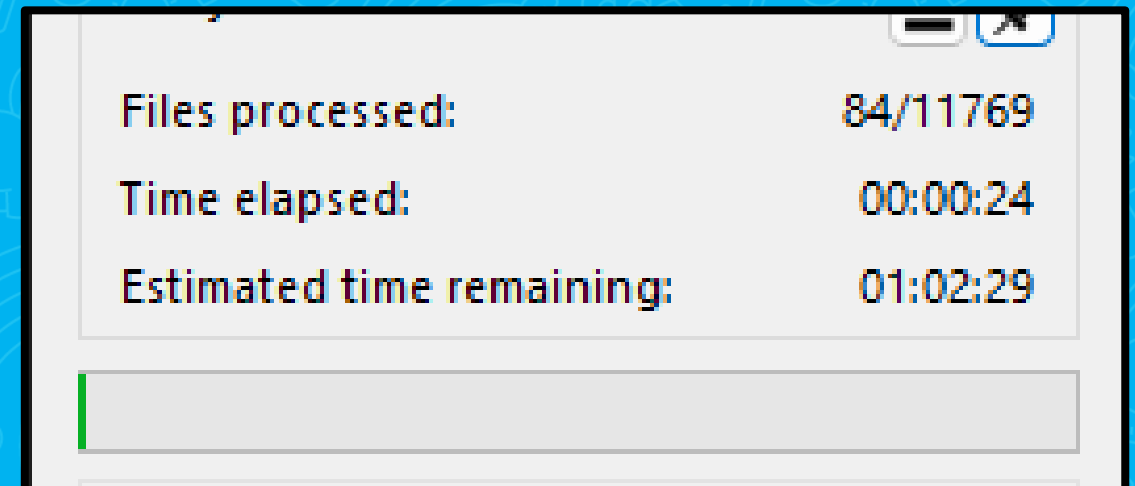
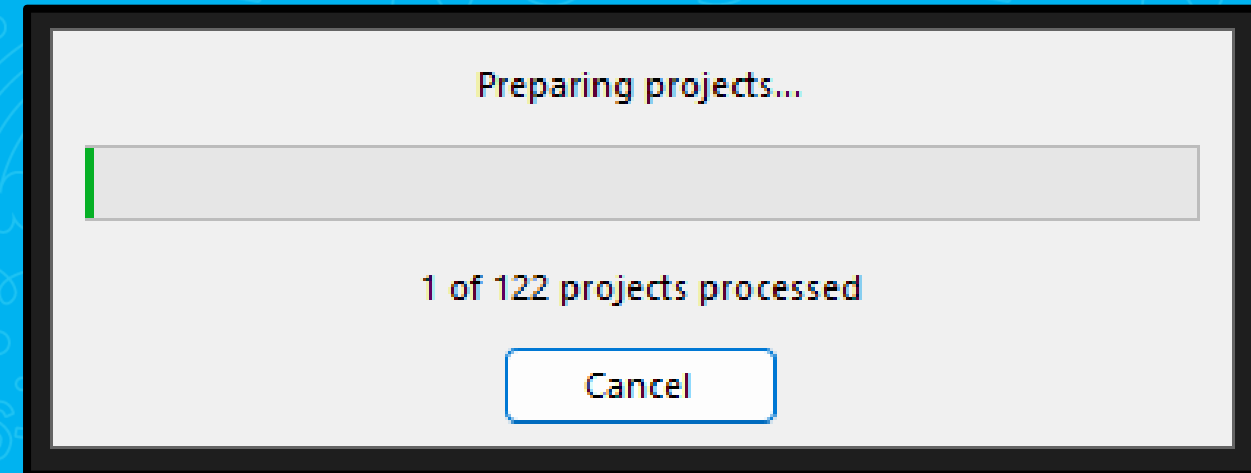
## «Избыточный» анализ

- Лишние файлы



## «Избыточный» анализ

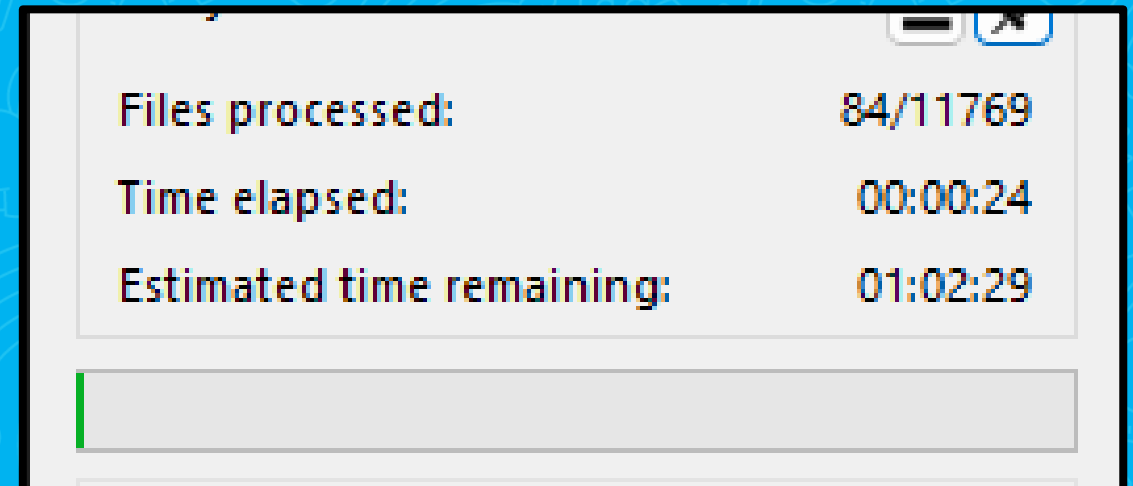
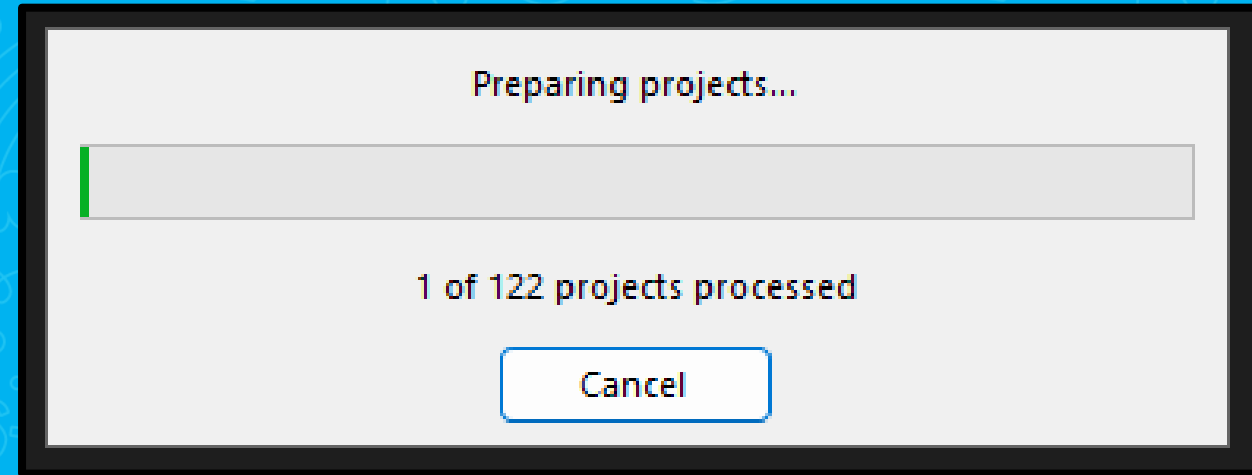
- Лишние файлы
- Внешние библиотеки/зависимости





## «Избыточный» анализ

- Лишние файлы
- Внешние библиотеки/зависимости
- Возвращаемся к настройке!
- Но не все так просто



# Не хватает диагностик

- Анализаторы не может покрыть все случаи и паттерны ошибок

Привет, спишь?)

Мне нужна диагностика которая найдет вот это:

[illegible]





## Дополнительные возможности настройки



## 5. MISRA

Show All ▾

- |       |   |   |
|-------|---|---|
| V2501 | MISRA. Octal constants should not be used.  | ✓ |
| V2502 | MISRA. The 'goto' statement should not be used.   | ✓ |
| V2503 | MISRA. Implicitly specified enumeration constants should be unique – consider specifying non-unique constants explicitly. | ✓ |
| V2504 | MISRA. Size of an array is not specified.   | ✓ |
| V2505 | MISRA. The 'goto' statement shouldn't jump to a label declared earlier.   | ✓ |

# ■ MISRA



## 5. MISRA

Show All ▾

V2501	MISRA. Octal constants should not be used.	<input checked="" type="checkbox"/>
V2502	MISRA. The 'goto' statement should not be used.	<input checked="" type="checkbox"/>
V2503	MISRA. Implicitly specified enumeration constants should be unique – consider specifying non-unique constants explicitly.	<input checked="" type="checkbox"/>
V2504	MISRA. Size of an array is not specified.	<input checked="" type="checkbox"/>
V2505	MISRA. The 'goto' statement shouldn't jump to a label declared earlier.	<input checked="" type="checkbox"/>

# ■ AUTOSAR

## 6. AUTOSAR

Show All ▾

V3501	AUTOSAR. Octal constants should not be used.	<input checked="" type="checkbox"/>
V3502	AUTOSAR. Size of an array is not specified.	<input checked="" type="checkbox"/>
V3503	AUTOSAR. The 'goto' statement shouldn't jump to a label declared earlier.	<input checked="" type="checkbox"/>
V3504	AUTOSAR. The body of a loop\conditional statement should be enclosed in braces.	<input checked="" type="checkbox"/>
V3505	AUTOSAR. The function with the 'atof/atoi/atol/atoll' name should not be used.	<input checked="" type="checkbox"/>
V3506	AUTOSAR. The function with the 'abort/exit/getenv/system' name should not be used.	<input checked="" type="checkbox"/>

# ■ MISRA

# ■ MISRA

5. MISRA		Show All ▾
V2501	MISRA. Octal constants should not be used.	<input checked="" type="checkbox"/>
V2502	MISRA. The 'goto' statement should not be used.	<input checked="" type="checkbox"/>
V2503	MISRA. Implicitly specified enumeration constants should be unique – consider specifying non-unique constants explicitly.	<input checked="" type="checkbox"/>
V2504	MISRA. Size of an array is not specified.	<input checked="" type="checkbox"/>
V2505	MISRA. The 'goto' statement shouldn't jump to a label declared earlier.	<input checked="" type="checkbox"/>

# ■ AUTOSAR

6. AUTOSAR		Show All ▾
V3501	AUTOSAR. Octal constants should not be used.	<input checked="" type="checkbox"/>
V3502	AUTOSAR. Size of an array is not specified.	<input checked="" type="checkbox"/>
V3503	AUTOSAR. The 'goto' statement shouldn't jump to a label declared earlier.	<input checked="" type="checkbox"/>
V3504	AUTOSAR. The body of a loop\conditional statement should be enclosed in braces.	<input checked="" type="checkbox"/>
V3505	AUTOSAR. The function with the 'atof/atoi/atol/atoll' name should not be used.	<input checked="" type="checkbox"/>
V3506	AUTOSAR. The function with the 'abort/exit/getenv/system' name should not be used.	<input checked="" type="checkbox"/>

7. OWASP		Show All ▾
V5001	OWASP. It is highly probable that the semicolon ';' is missing after 'return' keyword.	<input checked="" type="checkbox"/>
V5002	OWASP. An empty exception handler. Silent suppression of exceptions can hide the presence of bugs in source code during testing.	<input checked="" type="checkbox"/>
V5003	OWASP. The object was created but it is not being used. The 'throw' keyword could be missing.	<input checked="" type="checkbox"/>
V5004	OWASP. Consider inspecting the expression. Bit shifting of the 32-bit value with a subsequent expansion to the 64-bit type.	<input checked="" type="checkbox"/>
V5005	OWASP. A value is being subtracted from the unsigned variable. This can result in an overflow. In such a case, the comparison operation can potentially behave unexpectedly.	<input checked="" type="checkbox"/>

# ■ OWASP



# SonarQube







**Группа диагностик в PVS-Studio**

- Диагностики общего назначения
- Диагностики микро-оптимизаций
- Диагностики 64-битных ошибок
- Стандарт MISRA
- Диагностики, реализованные по запросам пользователей
- Проблемы при работе анализатора кода



**Тег в SonarQube**

- pvs-studio#ga
- pvs-studio#op
- pvs-studio#64
- pvs-studio#misra
- pvs-studio#cs
- pvs-studio#fails

 Projects Issues Rules Quality Profiles Quality Gates Administration More 



☆ Project / main  

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

 To benefit from more of SonarQube's features, [set up analysis in your favorite CI.](#) 


main

176k Lines of Code • Version **not provided** • [Set as homepage](#)

 Quality Gate 

**Passed**

Last analysis 5 minutes ago

 The last analysis has warnings. [See details](#)

New Code

Overall Code


Security

0 Open issues

0 H

0 M

0 L




Reliability

423 Open issues

108 H

169 M

146 L




Maintainability

0 Open issues

0 H


0 M

0 L



Accepted issues


0



Valid issues that were not fixed

Coverage


0.0%



On 40k lines to cover.

Duplications


1.5%





On 231k lines.



Security Hotspots


0





 [Projects](#) [Issues](#) [Rules](#) [Quality Profiles](#) [Quality Gates](#) [Administration](#) [More](#) 

☆ Project / main  

[Overview](#) [Issues](#) [Security Hotspots](#) [Measures](#) [Code](#) [Activity](#) Project Settings 

src/bridge/Utf8Ini.h

V781: The value of the 'len' index is checked after it was used. Perhaps there is a mistake in program logic.

src/bridge/bridgelist.h

V522: There might be dereferencing of a potential null pointer '(Type \*) listInfo->data'.

src/bridge/bridgemain.cpp



V576: Incorrect format. Consider checking the third actual argument of the 'scanf\_s' function. A pointer to the unsigned int type is expected.




V566: The integer constant is converted to pointer. Possibly an error or a bad coding style: (DWORD \*) (0x7FFE0000 + 0x260)

V1109: The 'IsBadWritePtr' function is deprecated. Consider switching to an equivalent newer function.

100 of 293 shown Show More

V781: Value of a variable is checked after it is used. Possible error in program's logic. Check lines: N1, N2. [pvs-studio-cxx-only:V781](#)


Line affected: L243 • Introduced: 8 years ago •  Vulnerability •  Major


☐ Open  ☐ Not assigned  cwe  ... +

Where is the issue?

Why is this an issue?

Activity

Project > src/bridge/Utf8Ini.h 

Open in IDE [See all issues in this file](#) 

↑

238 nukem@\_

239

240

241

242

243

244

245

246

247

248

249

250

251

252

↓

```
        return "";
        size_t pre = 0;
        while(str[pre] == ' ')
            pre++;
        size_t post = 0;
        while(str[len - post - 1] == ' ' && post < len)
            post++;
        auto sublen = len - post - pre;
        return sublen > 0 ? str.substr(pre, len - post - pre) : "";
    }

    static inline bool parseKeyValueLine(const std::string & line, std::string & key, std::string & value)
    {
        auto pos = line.find('=');
        key = trim(line.substr(0, pos));
```

V781: The value of the 'len' index is checked after it was used. Perhaps there is a mistake in program logic.

V781: The value of the 'post' index is checked after it was used. Perhaps there is a mistake in program logic.

## Форматы отчета

Формат	Расширение	Инструменты	Описание
PVS-Studio Log (Plog)	.plog	Visual Studio, SonarQube, Compiler Monitoring UI	Для Windows пользователей Visual Studio и SonarQube
JSON	.json	Visual Studio IntelliJ IDEA Rider CLion	Для пользователей плагинов PVS-Studio в IDE и SonarQube
SARIF	.sarif	Visual Studio, Visual Studio Code, есть визуализация в GitHub Actions	Универсальный формат отчёта статического анализатора
TaskList	.tasks	Qt Creator	Для работы с отчётом в Qt Creator
TaskList Verbose	.tasks	Qt Creator	Расширение формата TaskList с поддержкой отображения дополнительных позиций
CSV	.csv	Microsoft Excel LibreOffice Calc	Для просмотра предупреждений в табличном виде

Simple Html	.html	Email Client Browser	Для рассылки отчётов почтой
Full Html	Folder	Browser	Для просмотра предупреждений с навигацией по коду в браузере
Error File	.err	IDEs, Vim, Emacs, etc	Для просмотра отчётов в любом редакторе, поддерживающем формат вывода компилятора
Error File Verbose	.err	IDEs, Vim, Emacs, etc	Расширение формата Error File с поддержкой отображения дополнительных позиций
TeamCity	.txt	TeamCity	Для загрузки и просмотра предупреждений в TeamCity
MISRA Compliance	.html	Email Client Browser	Для проверки кода на соответствие стандартам MISRA
GitLab	.json	GitLab	Для просмотра предупреждений в формате GitLab Code Quality

# Форматы отчета

- Sarif

Универсальный формат отчёта статического анализатора.

Необходим для удобного обмена данными между различными инструментами статического анализа.

# Форматы отчета

- Sarif

Универсальный формат отчёта статического анализатора.

Необходим для удобного обмена данными между различными инструментами статического анализа.

- fullhtml

Это формат отчёта в виде HTML-файла.

При выборе этого формата plog-converter конвертирует сообщения анализатора и исходные файлы в html-файлы. Это позволяет просматривать отчёт анализатора в браузере с сортировкой по сообщениям и навигацией по коду.





# PVS-Studio Analysis Results

**Date:** 10/10/24 15:42:22  
**PVS-Studio Version:** 7.32.85062.1284  
**Total Warnings (GA):** 412

Show full paths ☒

Group	Projects	Location	Level	Code	Message
General Analysis	Translator	<a href="#">TranslatorWindow.xaml.cs:101</a>	Medium	<a href="#">V3063</a>	A part of conditional expression is always false if it is evaluated: e.Key == Key.Enter.
General Analysis	Translator	<a href="#">TranslatorWindow.xaml.cs:175</a>	Medium	<a href="#">V3063</a>	A part of conditional expression is always false if it is evaluated: e.Key == Key.Return.
General Analysis	Translator	<a href="#">DataGridHelper.cs:23</a>	Medium	<a href="#">V3080</a>	Possible null dereference. Consider inspecting 'presenter'.
General Analysis	ScreenToGif.Domain	<a href="#">DisplayDevices.cs:8</a>	Medium	<a href="#">V3117</a>	Constructor parameter 'filler' is not used.
General Analysis	ScreenToGif.Native	<a href="#">WindowMessageSink.cs:170</a>	Medium	<a href="#">V3083</a>	Unsafe invocation of event 'MouseEventReceived', NullReferenceException is possible. Consider assigning event to a local variable before invoking it.
General Analysis	ScreenToGif.Native	<a href="#">MemoryStatusEx.cs:17</a>	Medium	<a href="#">V3117</a>	Constructor parameter 'filler' is not used.
General Analysis	ScreenToGif.Native	<a href="#">TitlebarInfo.cs:26</a>	Medium	<a href="#">V3117</a>	Constructor parameter 'filler' is not used.
General Analysis	ScreenToGif.Native	<a href="#">WindowInfo.cs:63</a>	Medium	<a href="#">V3117</a>	Constructor parameter 'filler' is not used.
General Analysis	ScreenToGif.Util	<a href="#">FdatChunk.cs:36</a>	Medium	<a href="#">V3057</a>	The 1st argument 'length - 4' has a possibly negative value, but is expected to be non-negative inside method, in 'ms.Read(buffer, 0, (int)count)'.
General Analysis	ScreenToGif.Util	<a href="#">FdatChunk.cs:36</a>	Medium	<a href="#">V3171</a>	The parameter corresponding to the 1st argument 'length - 4' could reach the value of -4 inside method when used as the size of an array.
General Analysis	ScreenToGif.Util	<a href="#">GifPlainTextExtension.cs:57</a>	Medium	<a href="#">V3156</a>	The first argument of the 'GetString' method is not expected to be null. Potential null value: dataBytes.
General Analysis	ScreenToGif.Util	<a href="#">GifFile.cs:436</a>	High	<a href="#">V3022</a>	Expression '!IsFirstFrame    UseGlobalColorTable    UseFullTransparency' is always true.
General Analysis	ScreenToGif.Util	<a href="#">GifFile.cs:483</a>	High	<a href="#">V3022</a>	Expression '!IsFirstFrame    UseGlobalColorTable    UseFullTransparency' is always true.
General Analysis	ScreenToGif.Util	<a href="#">GrayscaleQuantizer.cs:24</a>	Medium	<a href="#">V3064</a>	Potential division by zero. Consider inspecting denominator '(MaxColorsWithTransparency - 1)'.
General Analysis	ScreenToGif.Util	<a href="#">NeuralQuantizer.cs:578</a>	Medium	<a href="#">V3106</a>	Possible negative index value. The value of 'closestNeuronIndex' index could reach -1.
General Analysis	ScreenToGif.Util	<a href="#">NeuralQuantizer.cs:579</a>	Medium	<a href="#">V3106</a>	Possible negative index value. The value of 'closestNeuronIndex' index could reach -1.
General Analysis	ScreenToGif.Util	<a href="#">LZWEncoder.cs:167</a>	Medium	<a href="#">V3117</a>	Constructor parameters 'width', 'height' are not used.
General Analysis	ScreenToGif.Util	<a href="#">Metadata.cs:12</a>	High	<a href="#">V3022</a>	Expression 'Content?.Length' is always not null. The operator '??' is excessive.
General Analysis	ScreenToGif.Util	<a href="#">MetadataContent.cs:16</a>	High	<a href="#">V3022</a>	Expression 'Content?.Length' is always not null. The operator '??' is excessive.
General Analysis	ScreenToGif.Util	<a href="#">ImageChannelData.cs:18</a>	High	<a href="#">V3022</a>	Expression 'Content?.Length' is always not null. The operator '??' is excessive.

```

26     var chunk = new FdatChunk
27     {
28         Length = length, //Chunk length, 4 bytes.
29         ChunkType = "fdAT" //Chunk type, 4 bytes.
30     };
31
32     using (var stream = new MemoryStream(array))
33     {
34         //Chunk details, 4 bytes + XX bytes.
35         chunk.SequenceNumber = BitHelper.ConvertEndian(stream.ReadUInt32());
36         chunk.FrameData = stream.ReadBytes(length - 4); //Minus 4 because that's the size of the sequence number.

```

↑ [V3171](#) The parameter corresponding to the 1st argument 'length - 4' could reach the value of -4 inside method when used as the size of an array.

↑ [V3057](#) The 1st argument 'length - 4' has a possibly negative value, but is expected to be non-negative inside method, in 'ms.Read(buffer, 0, (int)count)'.

```

37     }
38
39     return chunk;
40 }
41
42 internal void Write(Stream stream, bool writeAsIdat = true)
43 {
44     stream.WriteUInt32(BitHelper.ConvertEndian(Length)); //4 bytes.
45     stream.WriteByte(Encoding.ASCII.GetBytes(writeAsIdat ? "IDAT" : ChunkType)); //4 bytes.
46
47     if (!writeAsIdat)
48         stream.WriteUInt32(BitHelper.ConvertEndian(SequenceNumber)); //4 bytes.
49
50     stream.WriteByte(FrameData); //XX bytes.
51     stream.WriteUInt32(BitHelper.ConvertEndian(CrcHelper.Calculate(stream.PeekBytes(stream.Position - (Length + (writeAsIdat ? 4 : 8)), (int)Length + (writeAsIdat ? 4 :
52 }
53 }
54

```

## Возможности интеграции PVS-Studio

## IDE



Visual Studio



IntelliJ IDEA



Rider



CLion



Visual Studio Code



Qt Creator

Eclipse

## Игровые движки



Unreal Engine



Unity

## Качество кода



SonarQube

## Сборочные системы



MSBuild



CMake



Make



Ninja



Maven



Gradle

JSON Compilation

Database



## Embedded



Keil  $\mu$ Vision, DS-MDK



IAR Embedded



Workbench



Platform.io

QNX Momentics

TI ARM Code

Generation

## CI



Jenkins



TeamCity

## Облачные CI



CircleCI



Travis CI



GitLab

Azure DevOps

## Виртуализация



Docker



WSL

## Распределённая сборка



Incredibuild



# Q&A



Задавайте  
вопросы



Глеб Асламов