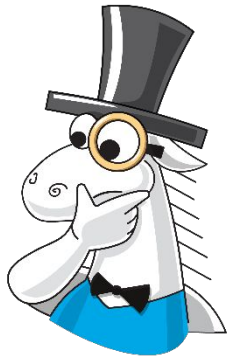
Великий и могучий C++

Андрей Карпов

karpov@viva64.com

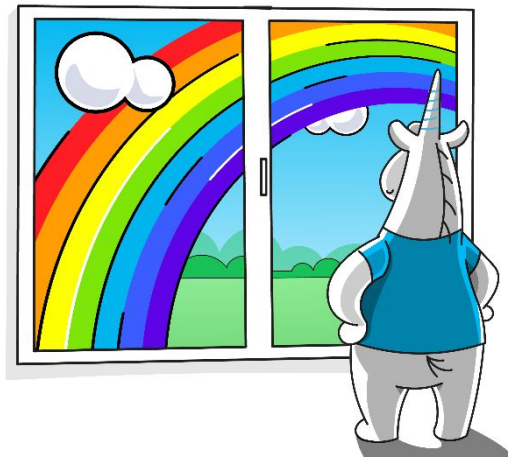
www.reddit.com/r/programming

- **SVG** Is Turing Complete
- A step-by-step guide to analyzing **JSON** (and other semi-structured datasets) with **SQL**
- “Performance Matters” by Emery Berger (Strange Loop 2019)
- Why are large companies so difficult to rescue (regarding bad internal technology)
- Practical Ways to Write Better **Javascript**
- From UNIX to Linux, a time lapse of 45 years, Hendrik Jan Thomassen
- Stack Overflow Trends – **Scala's** decline mirrors **Kotlin's** rise
- Maximize learnings from a **Kubernetes** cluster failure
- Why **Go** and not **Rust**?
- Q&A: How Google Implements Code Coverage at Massive Scale
- Hierarchical Instrumented Tracing with **Logback**
- Court of appeals rules web scraping doesn't violate anti-hacking law - **JavaScript** Web Scraping Guy
- Building **GraphQL** apis with **Django**



Просто каждый живёт в своей реальности

- C++ Russia x3 (Новосибирск, Москва, Санкт-Петербург)
- CoreHard (Минск, Минск)
- Meeting C++ (Берлин)
- CppCon (Colorado)
- C++Now (Colorado)
- Italian C++ Conference (Milan)
- emBO++ (Bochum)
- ....



Докладчик

- Карпов Андрей Николаевич, к.ф.-м.н.
- Технический директор ООО «СиПроВер»
- Microsoft MVP
- Intel Black Belt Software Developer
- Один из основателей проекта PVS-Studio

Несколько слов о нас

- Мы стоим на страже качества программного кода
- **PVS-Studio** выявляет ошибки и потенциальные уязвимости в коде программ, написанных на языках C, C++, C# и Java
- Активно участвуем в конференциях и пишем много статей, посвященных качеству кода

Жив ли C и C++? Живы программы для PDP!

- Мини-компьютер PDP-11

Жив ли C и C++? Да что там, жив IBM RPG!

- IBM RPG - язык программирования, синтаксис которого был изначально сходен с командным языком механических табуляторов компании IBM
- Широко использовался в 1960-х и 1970-х годах

IBM 1401

Однако, C и C++ это не удел старых систем

- C++ Applications - http://www.stroustrup.com/applications.html
- Adobe Photoshop
- Mozilla Firefox
- Thunderbird
- World of Warcraft
- Apple – OS X
- **Chrome (Дата выхода: 2 сентября 2008 г.)**

Жив ли C и C++?

- TIOBE Index: если сложить C и C++, то C/C++ будет лидером

Sep 2019	Sep 2018	Change	Programming Language	Ratings	Change
1	1		Java	16.661%	-0.78%
2	2		C	15.205%	-0.24%
3	3		Python	9.874%	+2.22%
4	4		C++	5.635%	-1.76%
5	6	^	C#	3.399%	+0.10%
6	5	v	Visual Basic .NET	3.291%	-2.02%
7	8	^	JavaScript	2.128%	-0.00%
8	9	^	SQL	1.944%	-0.12%
9	7	v	PHP	1.863%	-0.91%
10	10		Objective-C	1.840%	+0.33%

Резюме по C, C++

- Можно смело учить эти языки
- Есть и будет множество вакансий
- Увеличивается количество embedded-систем, где царствует C

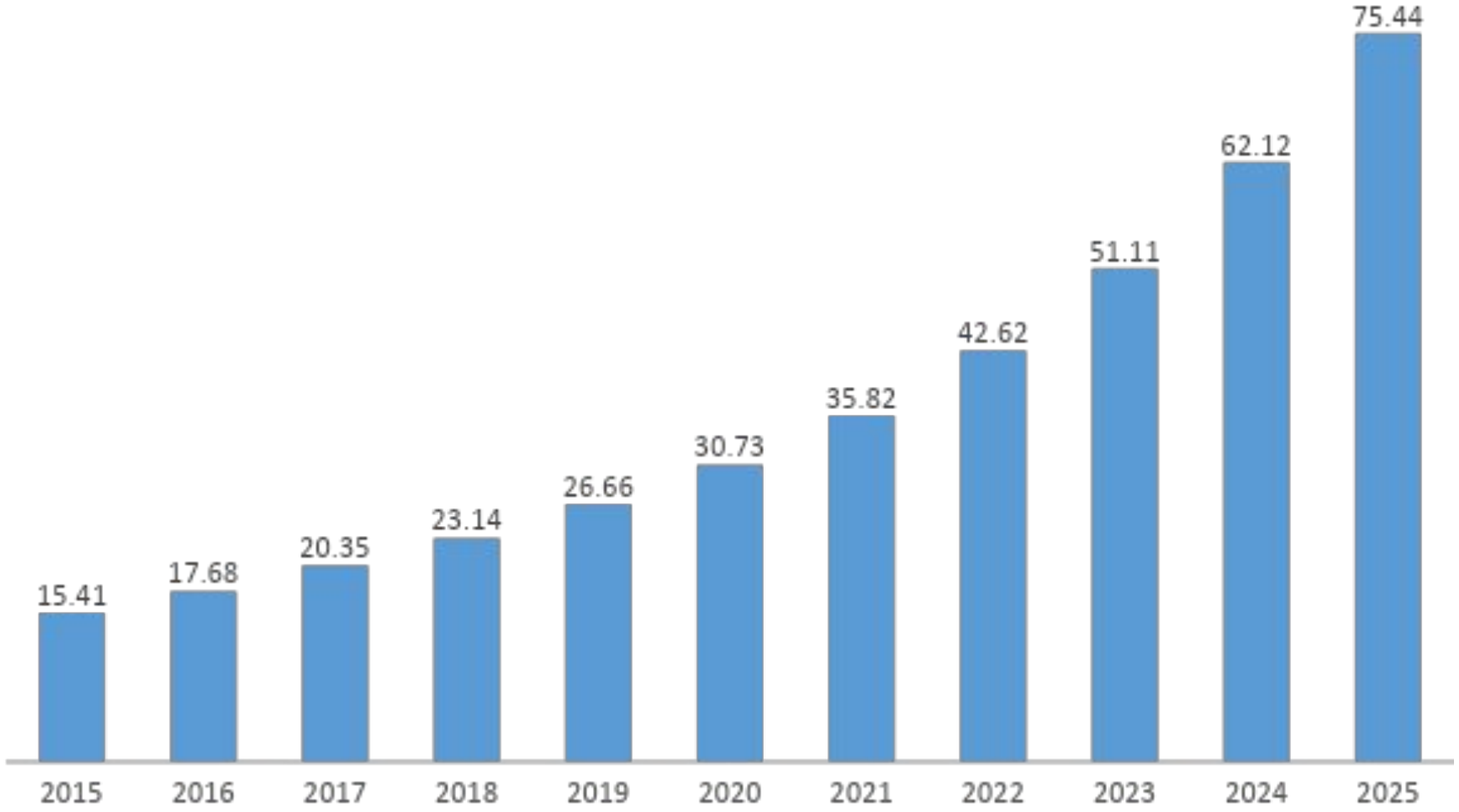
Темы:

- Embedded системы
- Немного о новшествах
- Зрелость инструментов разработки и увеличение скорости сборки проектов
- О тенденции безопасного программирования
- Разнообразие вспомогательного инструментария, позволяющего контролировать качество кода

Embedded: C и C++ вновь актуальны

- Embedded
- IoT (Интернет вещей)

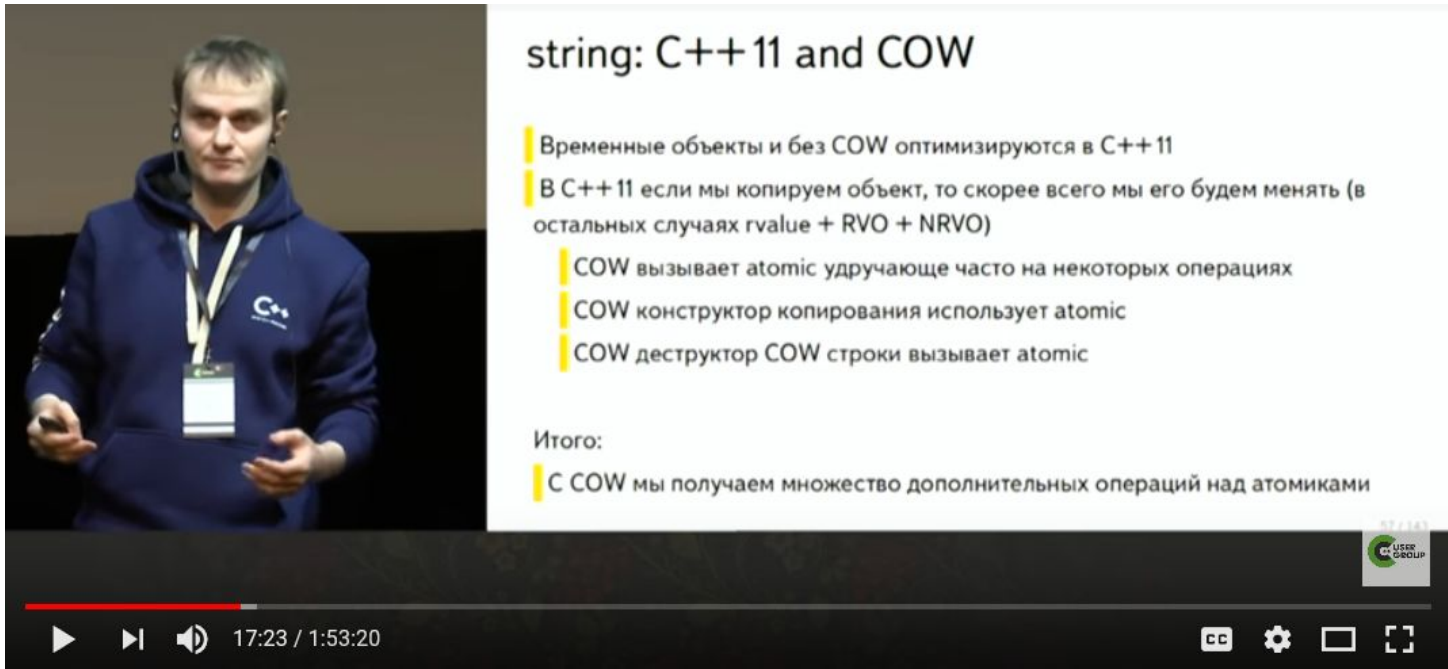
Прогнозируемые темпы роста IoT



Источник: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/

std::string

- C++ Russia 2017: Антон Полухин, Как делать не надо: C++ велосипедостроение для профессионалов https://youtu.be/rJWSSWYL83U
- vstring
- std::string



C++17: Constexpr if

```
template <typename T>
auto GetValue(T t)
{
  if constexpr (std::is_pointer<T>::value)
  {
    return *t;
  }
  else
  {
    return t;
  }
}
```

```
void foo()
{
  int v = 10;
  std::cout << GetValue(v) << '\n'; // 10
  std::cout << GetValue(&v) << '\n'; // 10
}
```

C++17: Инициализатор в if и switch

```
if (auto it = m.find(key); it != m.end())
{
   ....
}
```

C++17: новый атрибут [[fallthrough]]

```
switch (i)
{
case 10:
  f1();
  break;
case 20:
  f2();
  break;
case 30:
  f3();
  [[fallthrough]]; // Предупреждение будет подавлено
case 40:
  f4();
  break;
}
```

C++17: новый атрибут [[nodiscard]]

```
[[nodiscard]] int Sum(int a, int b)
{
  return a + b;
}

int main()
{
  Sum(5, 6); // Будет выдано предупреждение
             // компилятора/анализатора
  return 0;
}
```

C++17: Свёртка параметров шаблона (Fold expressions)

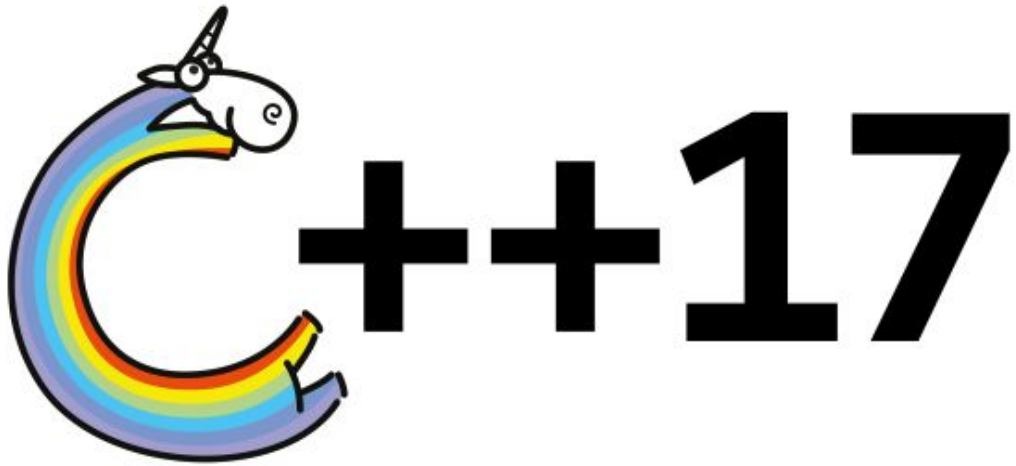
```cpp
template<typename... Args>
auto Sum(Args... args)
{
  return (args + ...);
}

int main()
{
  std::cout << Sum(1, 2, 3, 4, 5) << '\n'; // 15
  return 0;
}
```

C++17: удаленные возможности

- Удалены триграфы
- Ключевое слово register больше нельзя использовать как спецификатор переменной
- Удалены префиксный и постфиксный инкременты для типа bool
- Удален std::auto_ptr, вместо него стоит использовать std::unique_ptr

C++17: и так далее

- Рекомендую статью моего коллеги Егора Бредихина "C++17" https://www.viva64.com/ru/b/0533/



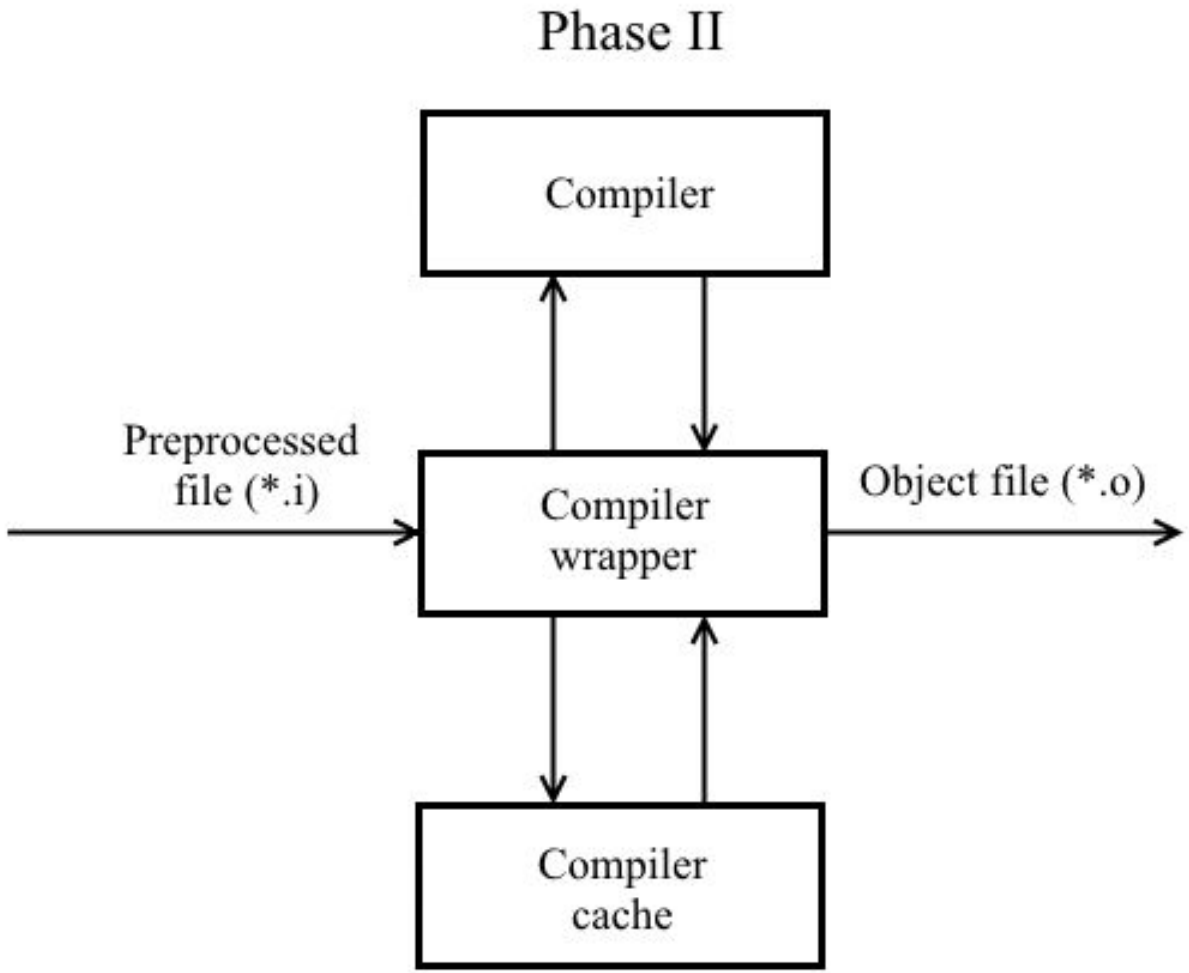
Ускорение сборки: распределённая компиляция

- **IncrediBuild** - https://www.incredibuild.com/
- **distcc** - https://github.com/distcc/distcc
- **Icecream** - https://github.com/icecc/icecream

Ускорение сборки: кэш компилятора

- При компиляции препроцессированного файла на основе его содержимого, флагов компиляции, вывода компилятора, вычисляется хэш-значение
- При повторной компиляции с теми же флагами неизмененного файла, из кэша будет взят уже готовый объектный файл и подан на вход компоновщика

Ускорение сборки: кэш компилятора



Ускорение сборки: кэш компилятора

- Для Unix-подобных систем:
  - **ccache** (GCC, Clang) - https://ccache.samba.org/
  - **cachecc1** (GCC) - http://cachecc1.sourceforge.net/
- Для Windows:
  - **clcache** (MSVC) - https://github.com/frerich/clcache
  - **cclash** (MSVC) - https://github.com/inorton/cclash

Ускорение сборки: ещё

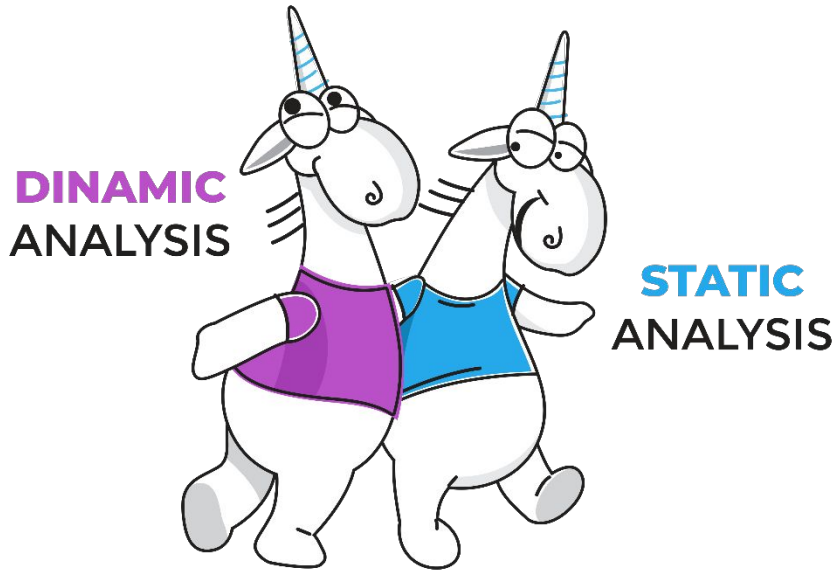
- Читайте статью моего коллеги Филиппа Хандельянца "Ускорение сборки C и C++ проектов" https://www.viva64.com/ru/b/0549/

Зрелость инструментария

- Компиляторы
- Среды для разработки
- Динамические анализаторы
- Статические анализаторы

Динамические анализаторы

- Классика:
  - Valgrind
  - BoundsChecker
  - Intel Parallel Inspector
- Новинки от Google:
  - AddressSanitizer
  - ThreadSanitizer
  - MemorySanitizer



Статические анализаторы кода

- Coverity
- Klocwork
- Parasoft
- PVS-Studio
- SonarQube
- "Имя им легион": List of tools for static code analysis https://en.wikipedia.org/wiki/List_of_tools_for_static_code_analysis

PVS-Studio

- C, C++, C#, Java
- Windows, Linux и macOS
- https://www.viva64.com/

- Бесплатные варианты лицензирования https://www.viva64.com/ru/b/0614/

```
static const int kDaysInMonth[13] = {
  0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31
};

bool ValidateDateTime(const DateTime& time) {
  if (time.year   < 1 || time.year   > 9999 ||
      time.month  < 1 || time.month  > 12   ||
      time.day    < 1 || time.day    > 31   ||
      time.hour   < 0 || time.hour   > 23   ||
      time.minute < 0 || time.minute > 59   ||
      time.second < 0 || time.second > 59) {
    return false;
  }
  if (time.month == 2 && IsLeapYear(time.year)) {
    return time.month <= kDaysInMonth[time.month] + 1;
  } else {
    return time.month <= kDaysInMonth[time.month];
  }
}
```

```
static const int kDaysInMonth[13] = {
  0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31
};

bool ValidateDateTime(const DateTime& time) {
  if (time.year   < 1 || time.year   > 9999 ||
      time.month  < 1 || time.month  > 12   ||
      time.day    < 1 || time.day    > 31   ||
      time.hour   < 0 || time.hour   > 23   ||
      time.minute < 0 || time.minute > 59   ||
      time.second < 0 || time.second > 59) {
    return false;
  }
  if (time.month == 2 && IsLeapYear(time.year)) {
    return time.month <= kDaysInMonth[time.month] + 1;
  } else {
    return time.month <= kDaysInMonth[time.month];
  }
}
```

```
static const int kDaysInMonth[13] = {
  0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31
};

bool ValidateDateTime(const DateTime& time) {
  if (time.year   < 1 || time.year   > 9999 ||
      time.month  < 1 || time.month  > 12   ||
      time.day    < 1 || time.day    > 31   ||
      time.hour   < 0 || time.hour   > 23   ||
      time.minute < 0 || time.minute > 59   ||
      time.second < 0 || time.second > 59) {
    return false;
  }
  if (time.month == 2 && IsLeapYear(time.year)) {
    return time.month <= kDaysInMonth[time.month] + 1;
  } else {
    return time.month <= kDaysInMonth[time.month];
  }
}
```

```
static const int kDaysInMonth[13] = {
  0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31
};

bool ValidateDateTime(const DateTime& time) {
  if (time.year   < 1 || time.year   > 9999 ||
      time.month  < 1 || time.month  > 12   ||
      time.day    < 1 || time.day    > 31   ||
      time.hour   < 0 || time.hour   > 23   ||
      time.minute < 0 || time.minute > 59   ||
      time.second < 0 || time.second > 59) {
    return false;
  }
  if (time.month == 2 && IsLeapYear(time.year)) {
    return time.month <= kDaysInMonth[time.month] + 1;
  } else {
    return time.month <= kDaysInMonth[time.month];
  }
}
```

Ошибка в проекте protobuf (Chromium)

time.day

Вопросы безопасности

- Актуальность (IoT, Embedded)
- Комплексные меры:
  - Стандарты кодирования
  - Покрытие кода
  - DAST
  - SAST

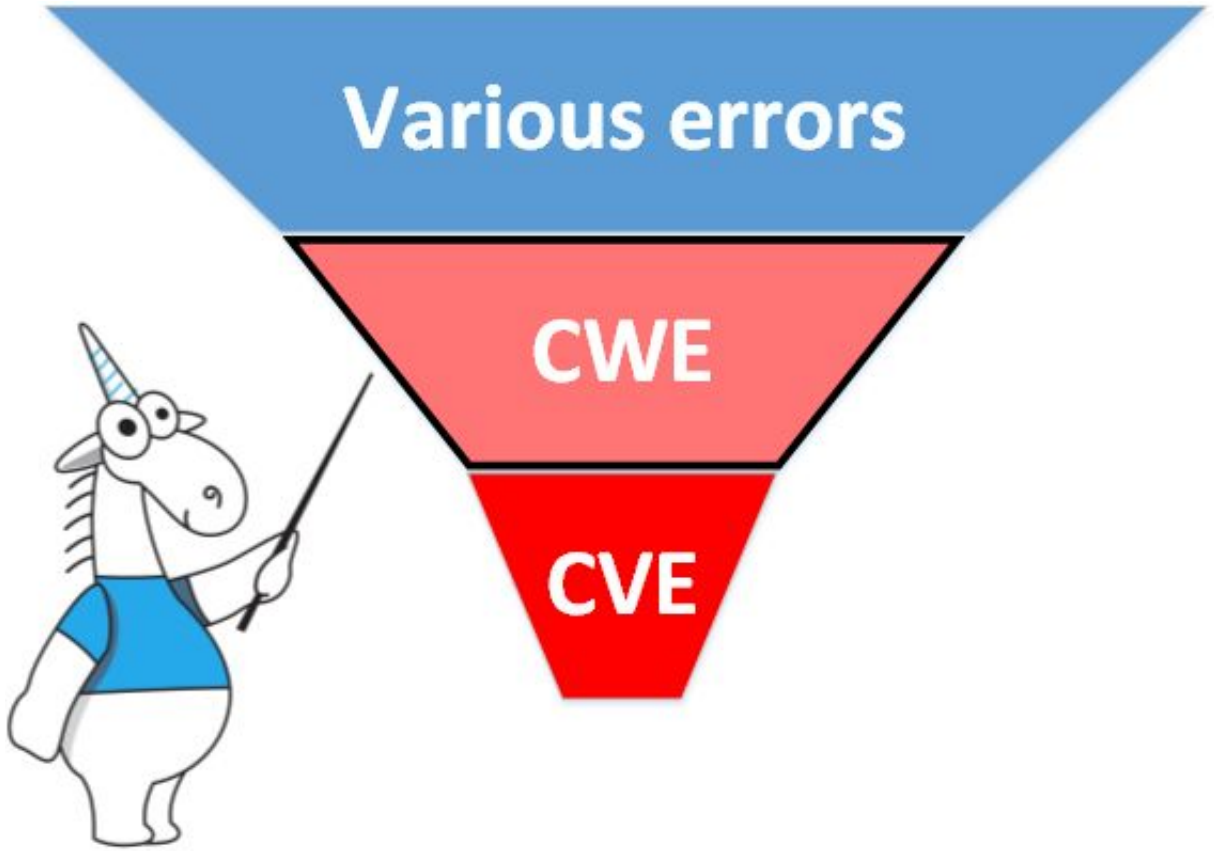
Стандарты кодирования

- Common Weakness Enumeration
- SEI CERT Coding Standards
- MISRA C, MISRA C++

- C++ Core Guidelines: https://github.com/isocpp/CppCoreGuidelines

CORE GUIDELINES

SAST - Static Application Security Testing



Заключение

- C и C++ однозначно стоит изучать
- Языки и инфраструктура активно развиваются

Ответы на вопросы

- Сайт PVS-Studio: https://www.viva64.com
- Контакты: CTO Андрей Карпов. karpov@viva64.com

- Подписывайтесь:
  - vk.com: Анализатор PVS-Studio - pvsstudio_rus
  - Instagram: @pvs_studio_unicorn