

# Security Code Reviews: Why You Should

**Presenter Name** – David Waters

**Event** – OWASP Day 2016

**Date** – 4<sup>th</sup> February 2016

## Presenter Introduction

- **David Waters**
- **Security Consultant @ Lateral Security**
- **16 Years Experience as a Developer**
- **~ 4 Years @ Google**
- **1 Year @ Google Security Team**
- **In the weekends likes long walks in the bush and planting trees on Motuihe Island.**

# Company Overview

## Company

- Lateral Security (IT) Services Limited
- Founded in April 2008 by Nick von Dadelszen and Ratu Mason (Both Directors)
- Auckland, Wellington Christchurch: ~20 highly specialised security consultants

## Services

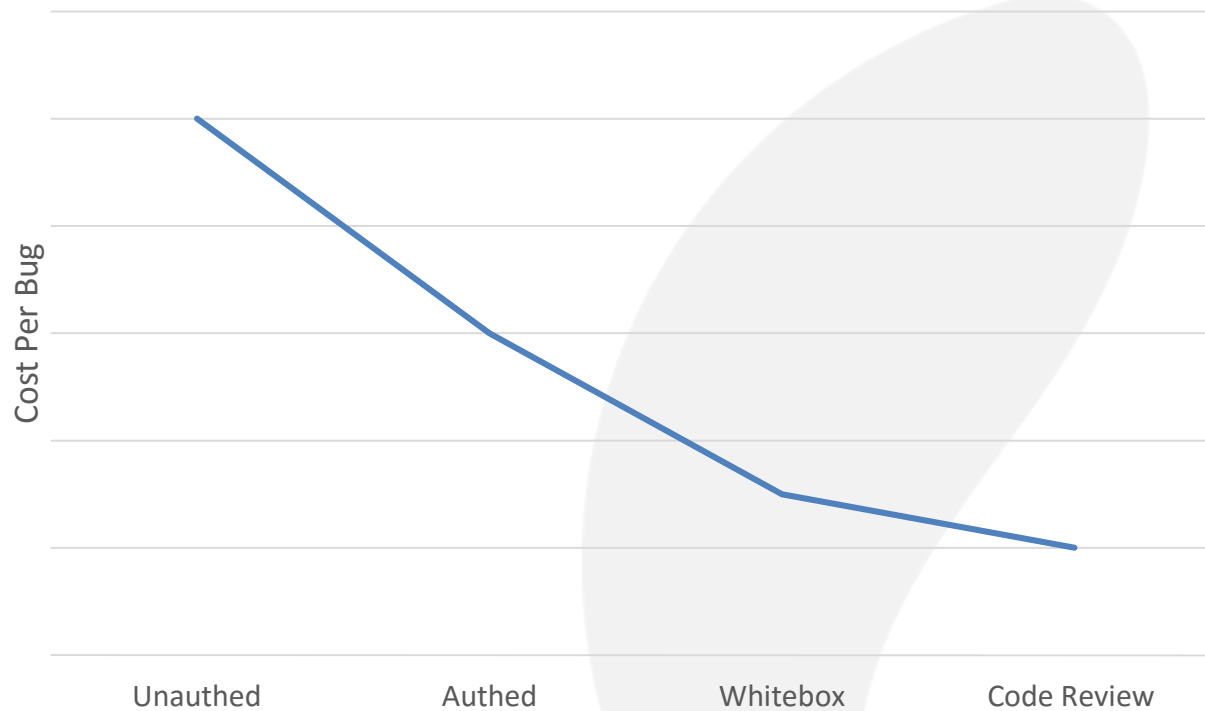
- Security testing (design & architecture, penetration testing, configuration, code reviews, security devices & controls, mobile apps)
- Security advisory (Lifecycle compliance & audit – ISO, PCI-DSS, NZISM, policy process development, threat modeling and risk assessment)
- Regular ongoing technical testing and assurance programs



## My Objectives

- **Awareness**
- **Persuasion**
- **Profit**

## Types Of Tests



## Why Do I Care

- **Huge Value**
- **Customer Surprise**
- **More Practitioners**

## Why Should You Care?

- **Better Targeted Penetration Testing**
- **Different Vulnerabilities Discovered**
- **Bad Patterns**
- **Future Security**

## Examples of Bad Code





```
1  /// <summary>Accepts a post request to change a user's password</summary>
2  [HttpPost]
3  public JsonResult Index(ChangePasswordModel changePasswordModel)
4  {
5      //this is login user code
6      string userCode = this.UserSession.UserCode;
7
8      // Trim leading and trailing spaces from the username and
9      // change password fields
10     if ( changePasswordModel.CurrentPassword == "null" )
11     {
12         userCode = changePasswordModel.ModifyUserCode;
13     }
14     ...
15     if ( changePasswordModel.CurrentPassword != "null" )
16     {
17         result = _userAccountServices.ChangePassword(userCode,
18             currentPassword, newPassword, confirmPassword);
19     }
20     else
21     {
22         result = _userAccountServices.UpdatePassword(userCode,
23             newPassword, confirmPassword);
24     }
25     ...
26 }
```



```
1  <%
2  ' asp file to create a single frameset
3  Frameset = Request( "Frameset" )
4  if Len( Frameset ) > 0 then
5      Response.Write( Frameset )
6  else
7      Title = Request( "Title" )
8      Url = Request( "Url" )
9      Scrolling = Request( "Scrolling" )
10     if Len( Title ) = 0 and Len( Scrolling ) = 0 then
11         Title = Label.GetGlobal("ModalDialogTitle","",Language)
12         ' Old style unescaped urls
13         Url = Request.ServerVariables( "QUERY_STRING" ), Len( Request.S
14     end if
15     Response.Write( "<HTML><TITLE>" & Title & "</TITLE><FRAMESET ROWS='*,0' F
16     if Scrolling = "Y" then
17         Response.Write( "<FRAME NORESIZE SRC='" & Url & "' NAME='Detached
18     else
19         Response.Write( "<FRAME NORESIZE SRC='" & Url & "' NAME='Detached
20     end if
21     if Request( "ZeroFrame" ) <> "N" then
22         Response.Write( "<FRAME NORESIZE SRC='/Darkblue.asp?bgColor=98C1E
23     end if
24     Response.Write( "</FRAMESET></HTML>" )
25 end if
26 %>
27
```



```
1 'VBScript ASP Unauthenticated
```

```
2  
3 UserId = Request( "UserId" )
```

```
4 IPAddress = Request( "IpAddress" )
```

```
5 ObrqPlacerGrp = Request( "PlacerGroupNumber" )
```

```
6 ObrqPlacerSch = Request( "PlacerGroupNumberSch" )
```

```
7 ' Extract form for display
```

```
8 If Len(UserId) > 0 then
```

```
9     sql = "select summaryform from weborderssummaryform where " & _
```

```
10         "UserId='" & UserId & "' " & _
```

```
11         "and IpAddress='" & IPAddress & "' " & _
```

```
12         "and obrq_placer_group='" & ObrqPlacerGrp & "' " & _
```

```
13         "and obrq_placer_group_sch='" & ObrqPlacerSch & "' "
```

```
14  
15 'response.write sql
```

```
16 sXML = GetDataCLOB( sql )
```

```
1 // C#
2
3 string _searchName = Request["Dictionary"];
4 string _queryInput = SQLFunctions.GetSafeSQL(Request["TypeAheadText"]);
5 _queryInput = _queryInput.Replace("%", "\\%");
6 string _queryField = Request["SearchField"].Trim().ToUpper();
7 string _searchFilter = Request["SearchFilter"].Trim();
8 string _multipleSelect = Request["MultipleSelect"].Trim().ToUpper();
9 string _sortBy = Request["SortBy"].Trim();
10 string _userAction = Request["UserAction"].Trim().ToUpper();
11 string _searchType = Request["SearchType"].Trim();
12 string _userLanguage = Request["Language"];
13 ...
14 string _queryArgument = _searchColumnName + " LIKE '";
15 if (_enableWildcardSearch)
16 {
17     _queryArgument += "%";
18 }
19 _queryArgument += _queryInput + "%' ESCAPE '\\'";
20
21 if (_searchFilter.Length > 0)
22 {
23     _queryArgument += (_queryArgument.Length > 0)
24                       ? " AND "
25                       : "";
26     _queryArgument += _searchFilter.Replace("|", "'");
27 }
28
```

```
1 'check the criteria can be evaluated
2 tcriteria = session("invgcriteria")
3 if len(tcriteria) > 0 then
4
5     fullCriteriaError = getCriteriaError(tcriteria)
6     if not fullCriteriaError = "" then
7         terror = 2
8     else
9         fullCriteriaError = "Ok"
10    end if
11
12 end if
13 ...
14 Function getCriteriaError( crit )
15
16     dim ignoreMe, criteriaError, regExMatches, regEx, regExMatch, keyWords
17
18
19     keyWords = array( "OR", "AND", "RiskRating", "PotentialConsequence", "Sa
20
21     criteriaError = "" 'assume success
22
23     'Temporarily switch off error handling
24     On Error Resume Next
25     ignoreMe = eval(crit) ' Try to evaluate to check syntax
26     If Err.Number <> 0 then criteriaError = err.Description 'Check for error
27     On Error Goto 0
28
29
30     if criteriaError = "" then 'syntax ok, check for valid variables
```



```
1 'Preserve Password
2 If Not Sender.FormSubmitted Then
3     Sender.password.Shadow.Value = CCEncryptString(MD5UsersPassword,
4     CCS_ENCRYPTION_KEY_FOR_COOKIE)
5     Sender.password.Value = ""
6 End If
7 .....
8 Function CCEncryptString(inputString, key)
9     CCEncryptString = CCBytesToHex(CCCipherEnDeCrypt(inputString, key))
10 End Function
11
12 Function CCCipherEnDeCrypt(inputString, key)
13     dim temp, a, i, j, k, crypted
14     dim result()
15     i = 0
16     j = 0
17     CCCipherInit key
18     For a = 1 To Len(inputString)
19         i = (i + 1) Mod 256
20         j = (j + cipherbox(i)) Mod 256
21         temp = cipherbox(i)
22         cipherbox(i) = cipherbox(j)
23         cipherbox(j) = temp
24         k = cipherbox((cipherbox(i) + cipherbox(j)) Mod 256)
25         crypted = Asc(Mid(inputString, a, 1)) Xor k
26         redim preserve result(a)
27         result(a-1) = crypted
28     Next
29     CCCipherEnDeCrypt = result
30 End Function
```



```
1 public Foo.User Login(long key, string email, string password, string version = "", string url = "", bool se
2 {
3
4     password = password.Replace("'", "");
5     email = email.Replace("'", "");
6
7     if (sendPassword)
8     {
9         foreach (Foo.User u in Common.GetObjectList<Foo.User>("select u.* from [User] u inner join Body b on
10             " where len(USER_EMAIL)>0 and len(USER_PASSWORD)>0 and USER_EMAIL='" + email + "'"))
11         {
12             try
13             {
14                 System.Net.Mail.MailMessage m = new System.Net.Mail.MailMessage();
15
16                 m.From = new System.Net.Mail.MailAddress("noreply@foo.co.nz", "Foo");
17                 m.To.Add(u.USER_EMAIL);
18                 m.Subject = "Foo Password Request";
19                 m.Body = u.USER_PASSWORD;
20
21                 Common.SendEmail(m);
22             }
23             catch { }
24         }
25         throw new Exception("Your password has been sent to " + email);
26     }
27     ....
28
```



```
1 public User Login(long key, string email, string password, string version = "", string url = "", bool sendP
2 {
3     string k1 = Convert.ToString((DateTime.Today.Month * (DateTime.Today.Day - 1) * 1000 + 123));
4     string k2 = Convert.ToString(DateTime.Today.Month * DateTime.Today.Day * 1000 + 123);
5     string k3 = Convert.ToString((DateTime.Today.Month * (DateTime.Today.Day + 1) * 1000 + 123));
6
7     StringBuilder sql = new StringBuilder();
8
9     if (this.Session == null)
10         this.Session = new Foo.Session();
11
12     if (password.Length == 0)
13         password = "1";
14
15     ...
16     isStaff = Common.ExecuteSQL<bool>("select case when BODY_SEQ=1 then 1 else 0 end from [User] where USER
17
18     if (isStaff && (email == k1 || email == k2 || email == k3)) // must be previously logged in as an Foo U
19     {
20         this.User = Common.GetObject<User>(Common.ParseInteger(password, true));
21         this.User.Body = Common.GetObject<Body>("select * from [Body] where BODY_SEQ=" + this.User.BODY_SEQ
22     }
23     else
24     {
25         // look up credentials database
26         this.User = Common.GetObjectSP<AutoPlay.User>("RoleManagement_GetUser", new List<SqlParameter>() {
27         this.User.Body = Common.GetObject<AutoPlay.Body>(this.User.BODY_SEQ);
28     }
29     //}
30
31     if (this.User.USER_SEQ == 0)
32         throw new Exception("Invalid Email/Password");
33
```



```
1 <?php
2 include_once('common.php');
3
4 $action_id = $_REQUEST['action_id'];
5 if (isset($_REQUEST['confirmed'])) {
6     $confirmed = $_REQUEST['confirmed'];
7 }
8 if ((!isset($confirmed)) or empty($confirmed) or ($confirmed != 'yes'))
9 {
10     echo "<html><p><a href='example.php?action_id={$action_id}&confirmed=yes'>Archive (I know what I
11         am doing)</a></html>";
12     exit();
13 }
14 $query = "SELECT at.base_directory
15           FROM action_type at, action a
16           WHERE a.action_id = {$action_id}
17           AND a.action_type_id = at.action_type_id";
18 $result = db_query($query, $connection);
19 if (!$row = db_fetch_array($result)) {
20     die('Could not find base_directory for this action type');
21 }
22 // Keep this on one line otherwise BASH will think it's two commands
23 $cmd = "./admin/shell_scripts/archive_action.sh {$action_id} {$userName} {$password} {$databaseName}
24         {$archive_directory} {$document_directory} {$mysql_bin} {$db_hostname} {$db_port}";
25 $output = array();
26 $return_code = 0;
27 exec($cmd, $output, $return_code);
28 if ($return_code != 0)
29 {
30     echo "<html><head><link href='form.css' rel='stylesheet' type='text/css'></head><body>";
31     echo "{$cmd}<br/>Return Code = {$return_code}<br />";
32     foreach ($output as $msg){echo "{$msg}<br />";}
33     echo "</body></html>";
34     exit();
35 }
36 ?>
```

# Questions and Contacts



**Presentation Download**  
[www.lateralsecurity.com/  
presentations](http://www.lateralsecurity.com/presentations)

## **Lateral Security (IT) Services Limited**

### **Wellington**

69 The Terrace (level 5, Gleneagles House)  
PO Box 8093, Wellington 6011, New Zealand  
Phone: +64 4 4999 756  
Email: [sas@lateralsecurity.com](mailto:sas@lateralsecurity.com)

### **Auckland**

53 High Street (level 1)  
PO Box 7706, Auckland, New Zealand  
Phone: +64 9 3770 700  
Email: [sas@lateralsecurity.com](mailto:sas@lateralsecurity.com)

### **Christchurch**

36 Byron Street (level 1)  
Sydenham 8023, Christchurch, New Zealand  
Phone: +64 35950387  
Email: [sas@lateralsecurity.com](mailto:sas@lateralsecurity.com)