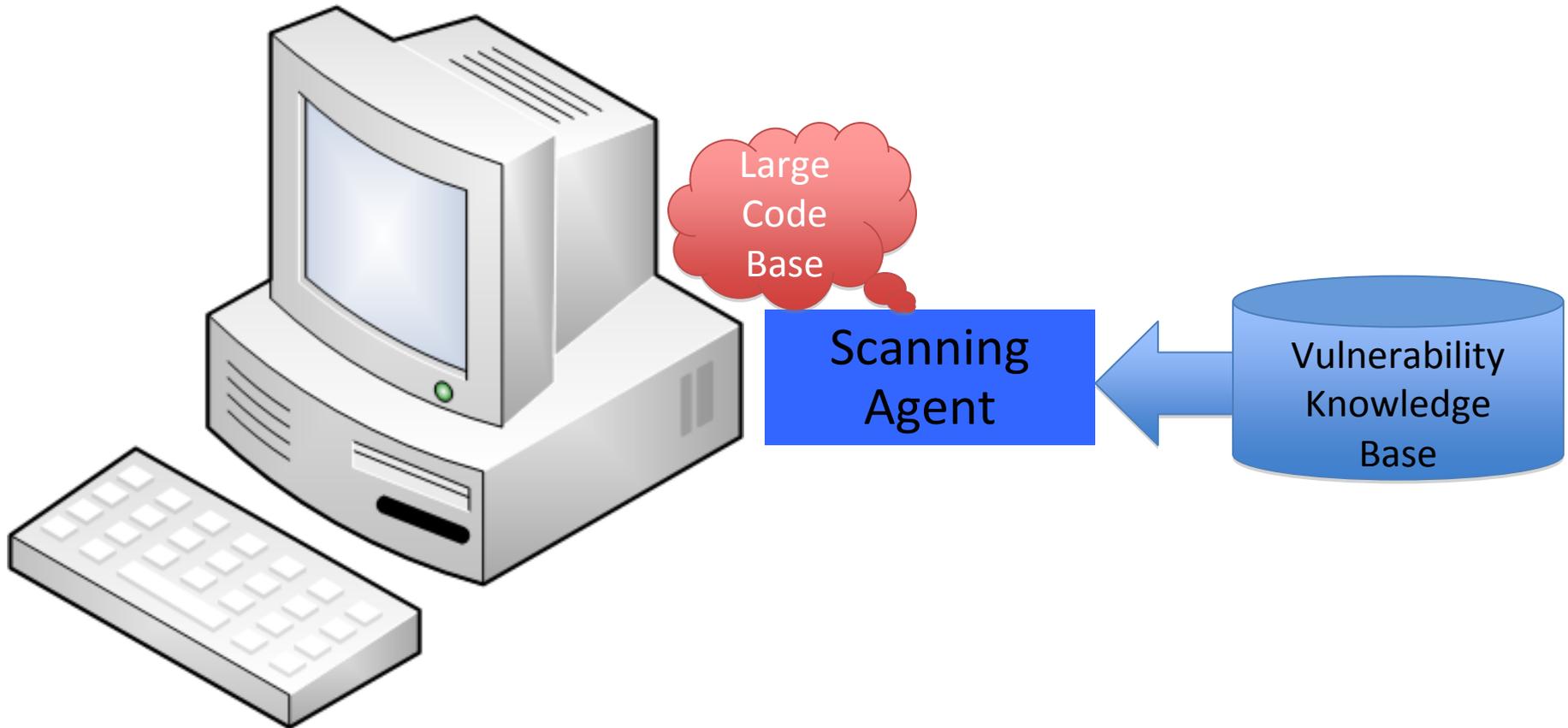


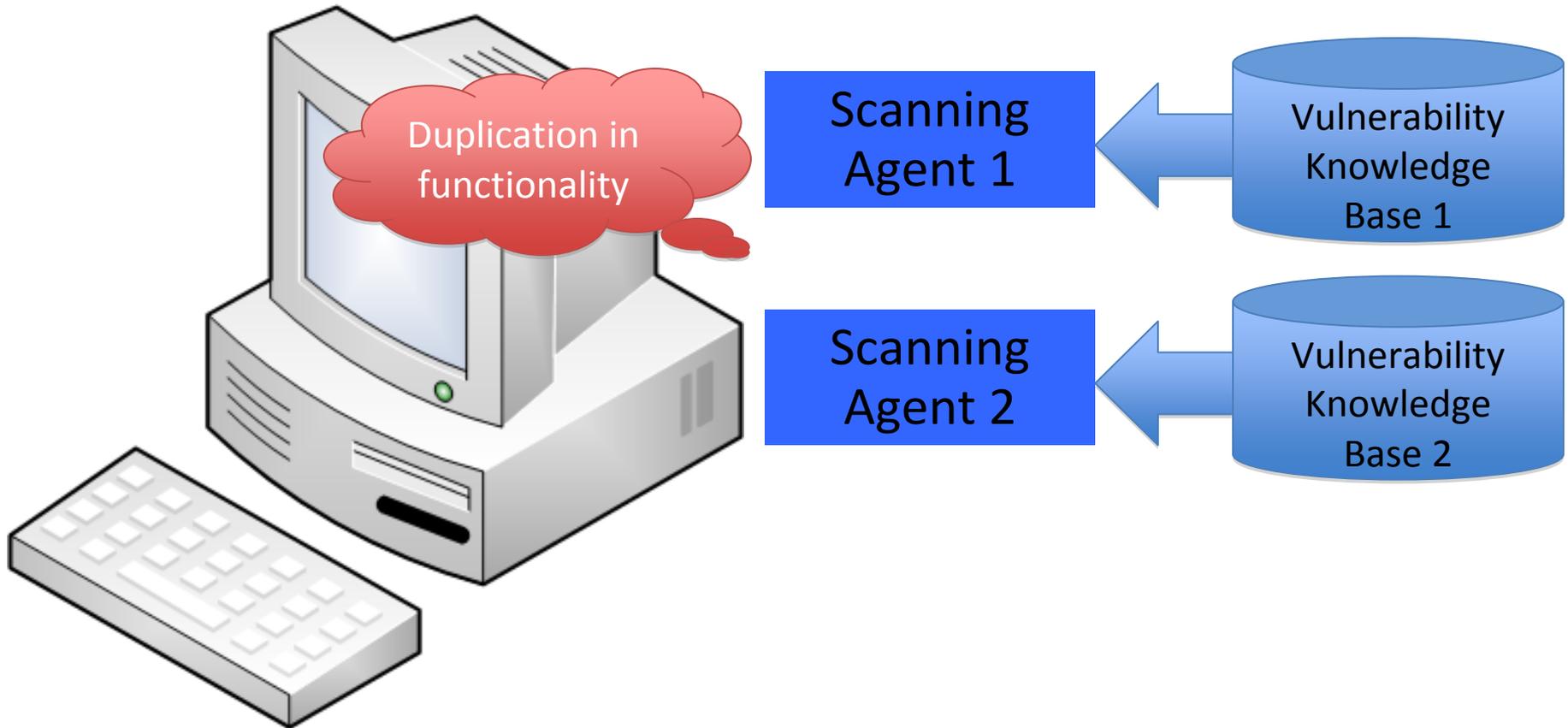
A host-based security assessment architecture for industrial control systems

Abhishek Rakshit and **Xinming (Simon) Ou**
Kansas State University

Host-based Vulnerability Scanner



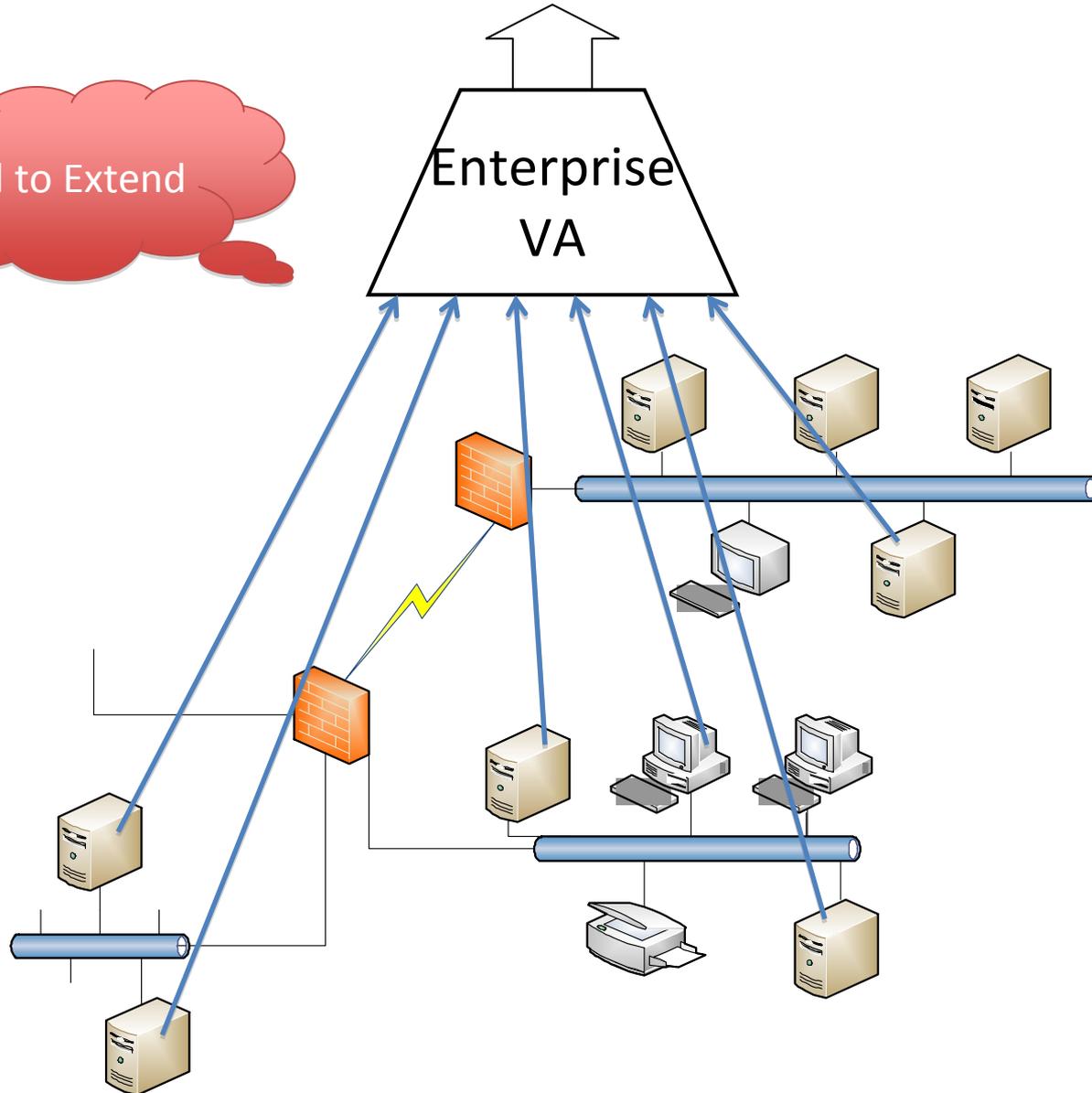
Host-based Vulnerability Scanner



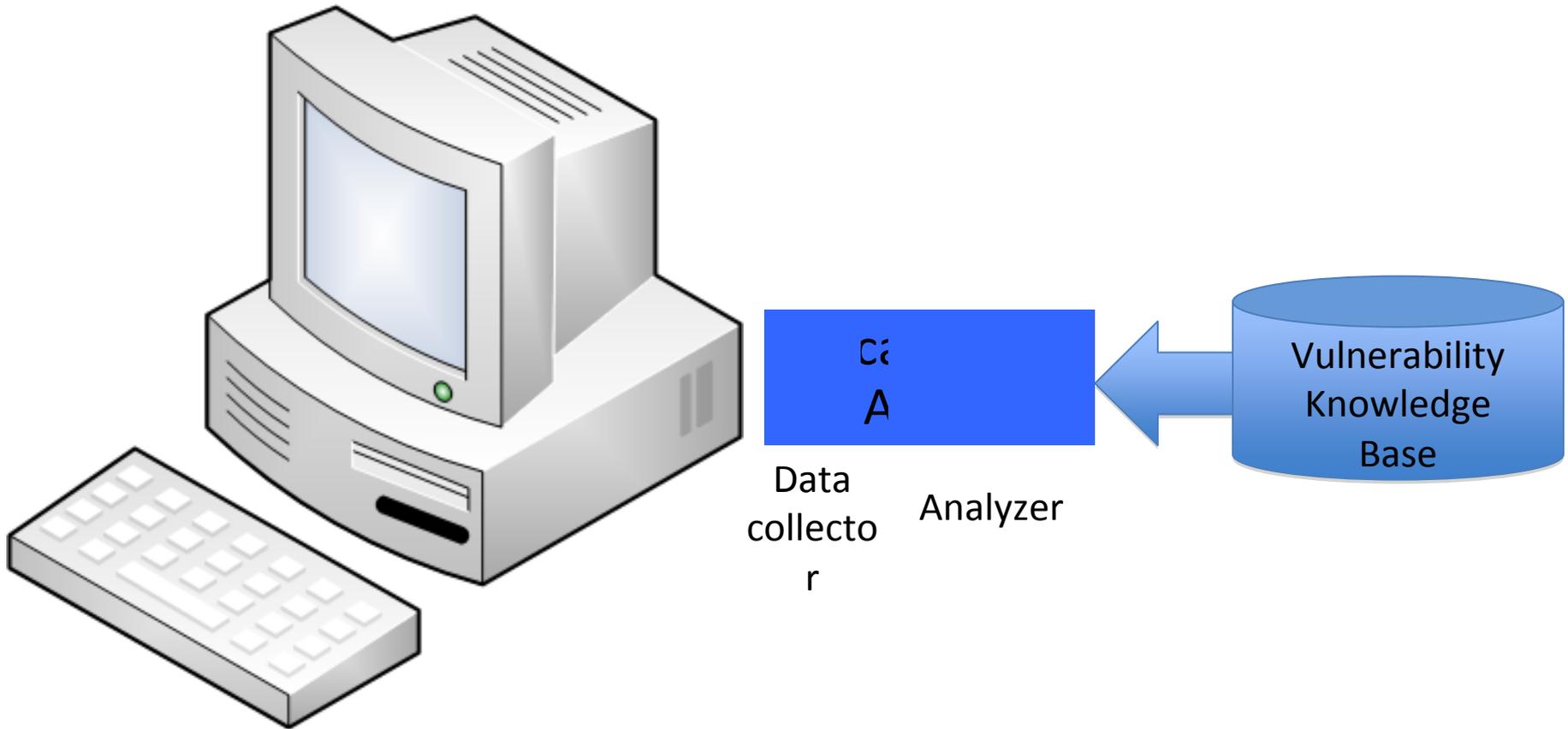


Global vulnerability assessment

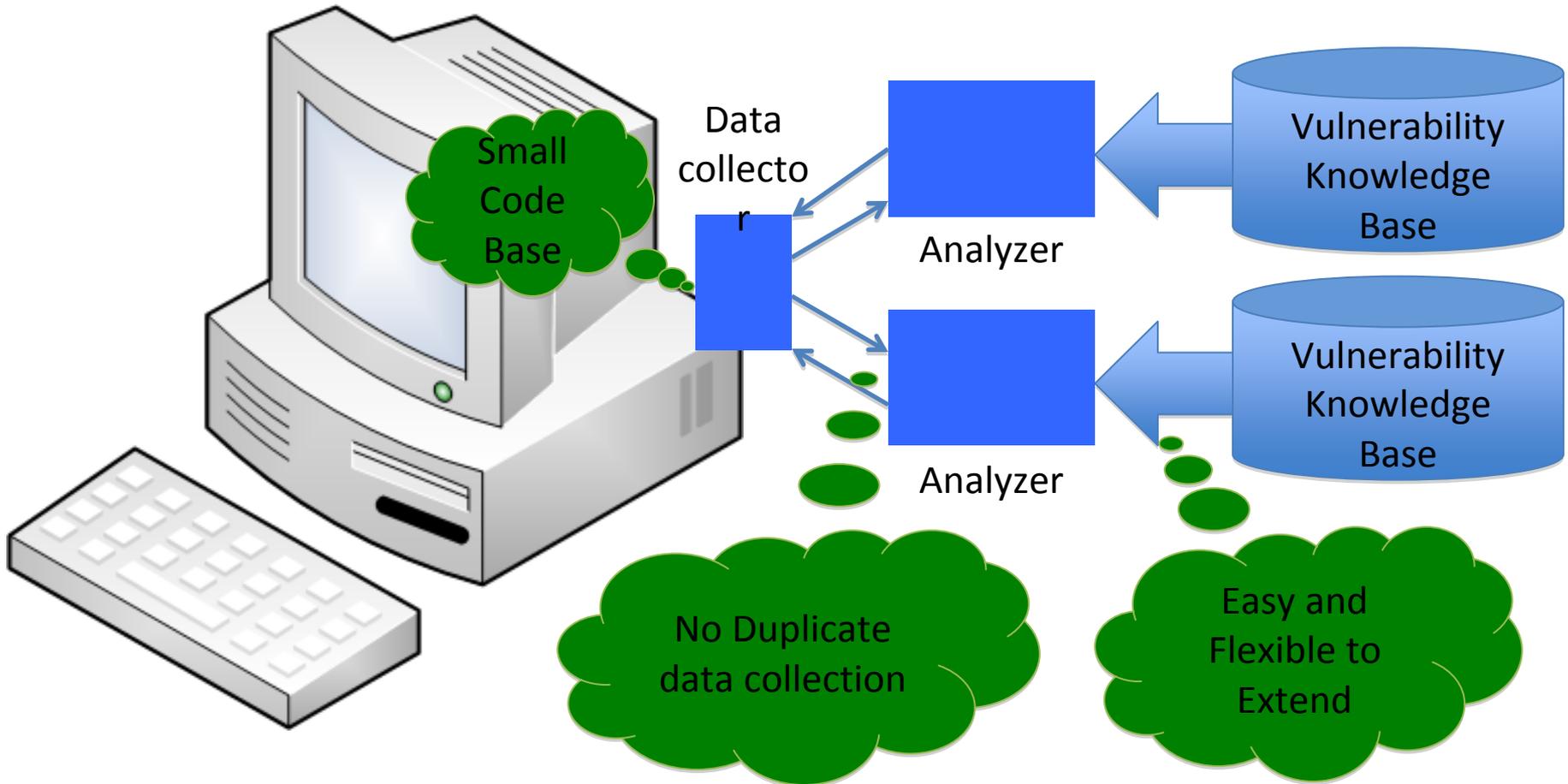
Hard to Extend



Proposed Architecture



Proposed Architecture



Case Study — OVAL Scanner

Title: Microsoft Outlook Advanced Find Vulnerability

Definition Id: oval:org.mitre.oval:def:153

CVE ID: 2007-0034

Definition Synopsis:

Outlook 2000

Outlook 2000 is installed

AND the version Outllib.dll is less than 9.0.0.8954

OR Outlook 2002

Outlook 2002 is installed

AND the version of Outllib.dll is less than 10.0.6822.0

OR Outlook 2003

Outlook 2003 is installed

AND the version of Outllib.dll is greater than 11.0.8118.0

OVAL XML Vulnerability Definition

```
<objects>
  <registry_object id="oval:org.mitre.oval:obj:670">
    <hive>HKEY_LOCAL_MACHINE</hive>
    <key>SOFTWARE\Microsoft\Office\9.0\Outlook\InstallRoot</key>
    <name>Path</name>
  </registry_object>
  <file_object id="oval:org.mitre.oval:obj:97">
    <path var_ref="oval:org.mitre.oval:var:728" />
    <filename>Outllib.dll</filename>
  </file_object>
</objects>
<states>
  <registry_state id="oval:org.mitre.oval:ste:804">
    <value operation="pattern match">.*\\[Oo][Ff][Ff][Ii][Cc][Ee][\\9].*</value>
  </registry_state>
  <file_state id="oval:org.mitre.oval:ste:160">
    <version datatype="version" operation="less than">9.0.0.8954</version>
  </file_state>
</states>
```

Data Collector for Registry Entries

```
regedit -e hcr.txt "HKEY_CLASSES_ROOT"
```

```
regedit -e hcu.txt "HKEY_CURRENT_USER"
```

```
regedit -e hlm.txt "HKEY_LOCAL_MACHINE"
```

```
regedit -e hcc.txt "HKEY_CURRENT_CONFIG"
```

```
regedit -e hu.txt "HKEY_USERS"
```

Data Collector for File Version No.

```
const ForReading = 1
const ForWriting = 2

strInput = "path.txt"
strOutput = "version_no.pl"
Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objInFile = objFSO.OpenTextFile(strInput, ForReading)

Set objOutFile = objFSO.CreateTextFile(strOutput)
objOutFile.close

Set objOutFile = objFSO.OpenTextFile(strOutput, ForWriting)

Do Until objInFile.AtEndOfStream
    strText = objInFile.ReadLine
If objFSO.FileExists(strText) = True Then
    strAttr = objFSO.GetFileVersion(strText)
    strText = Replace(strText, "\", "\\")
    strText = Replace(strText, "\\\\", "\\")
    strOut = "file_attr(" & """" & strText & """" & ", " & """" & strAttr & """" & ")."
    objOutFile.WriteLine(strOut)
End If
Loop

objInFile.Close
objOutFile.Close
```

21 Lines of Code



Comparison with MITRE's reference implementation of OVAL interpreter

	Reference impl.	Our impl.
Code Base (loc)	About 35,000	Less than 30
Running Time	Around 3 min	Around 2.5 min

Data collector runs on the target host. The analyzer runs on a dedicated Linux machine. The analyzer and target host communicate through SSH.

Conclusion

- It is feasible to separate configuration data collection and analysis in host-based vulnerability assessment
- The separation brings the benefit of increased trustworthiness in privileged code and flexibility in deployment and extension