

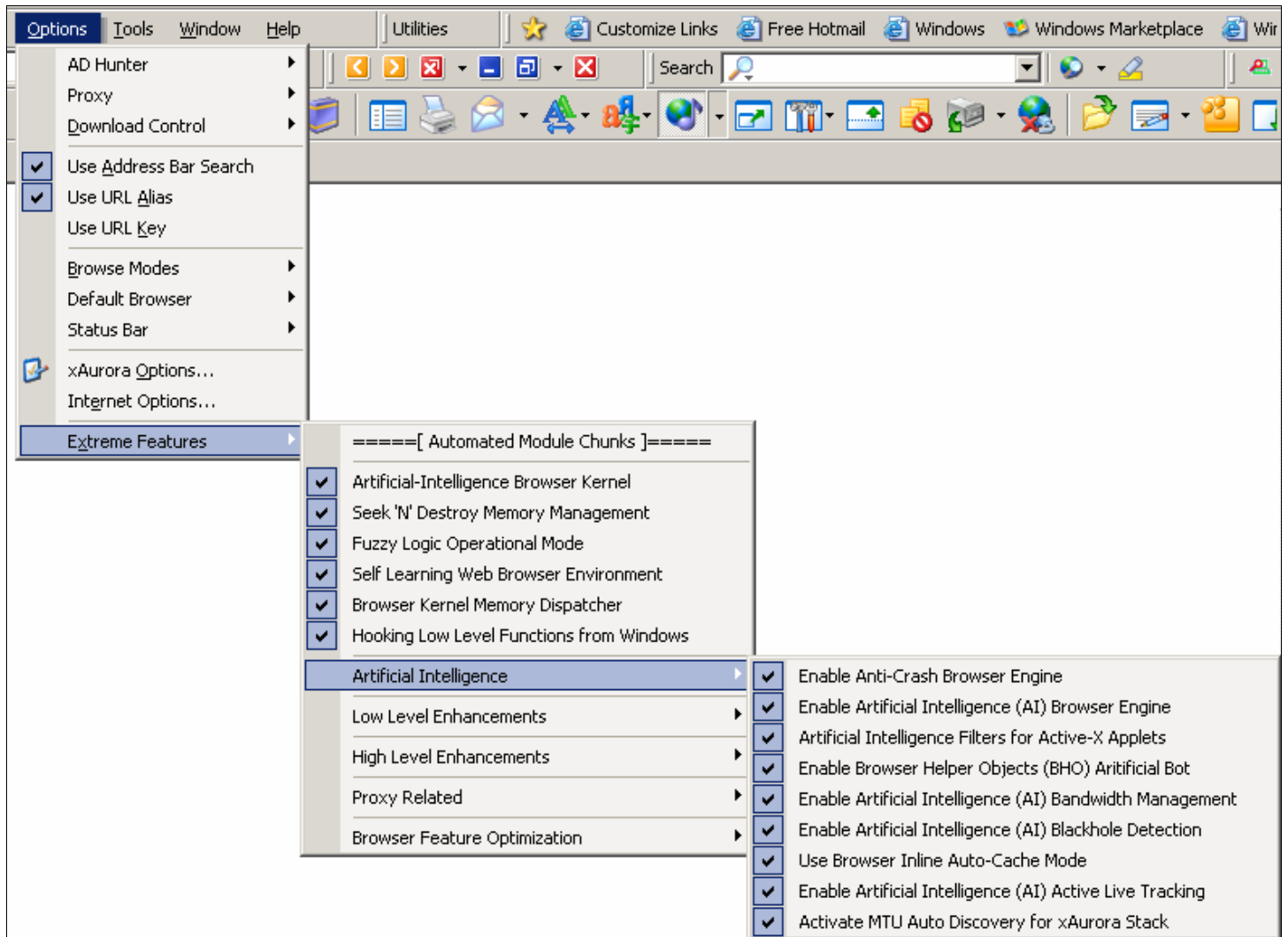
Reply to Mr. Gotaimbara

Extreme Features of xAurora (Extreme Features Menu)

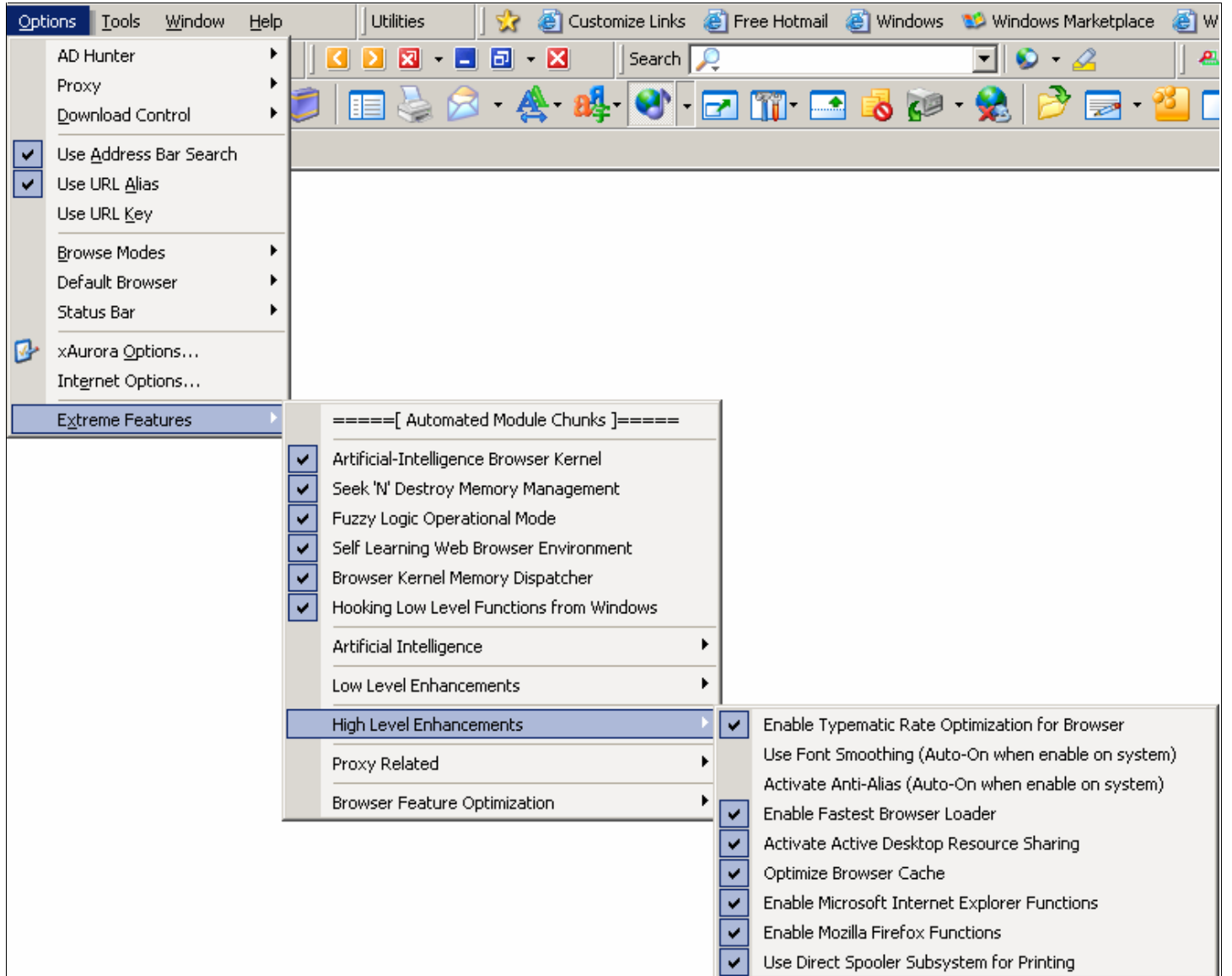
Response To : Mr. Prasan Hettiarchchi, Mr. Thathagatha, Mr. Kalinga Athulathmudali and Mr. FallenZeraphine

xAurora EXTREME FEATURES – Visual & Conceptual Approach

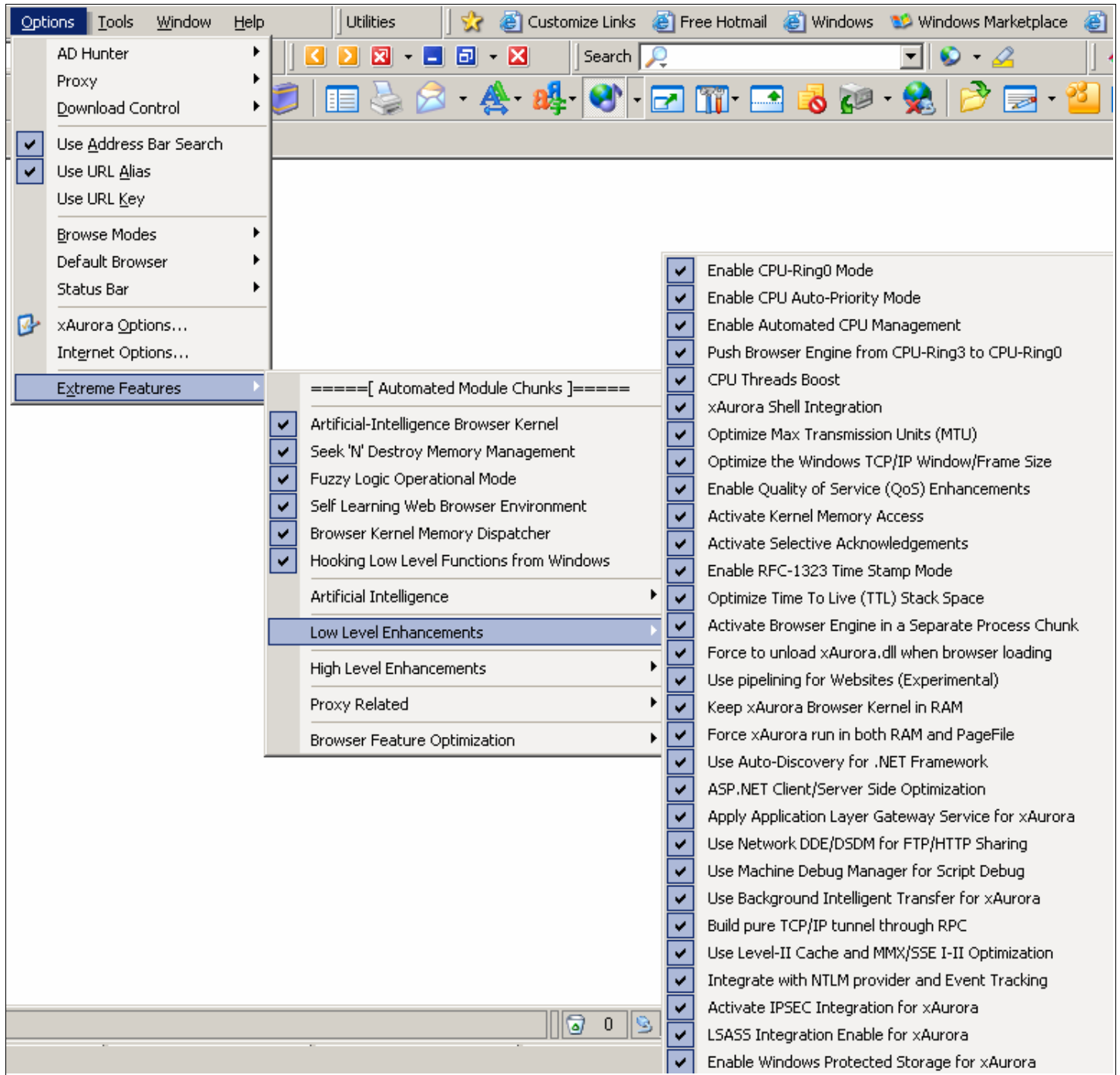
Extreme Features which Maxthon or any other browsers DOES NOT STEPPED OUT



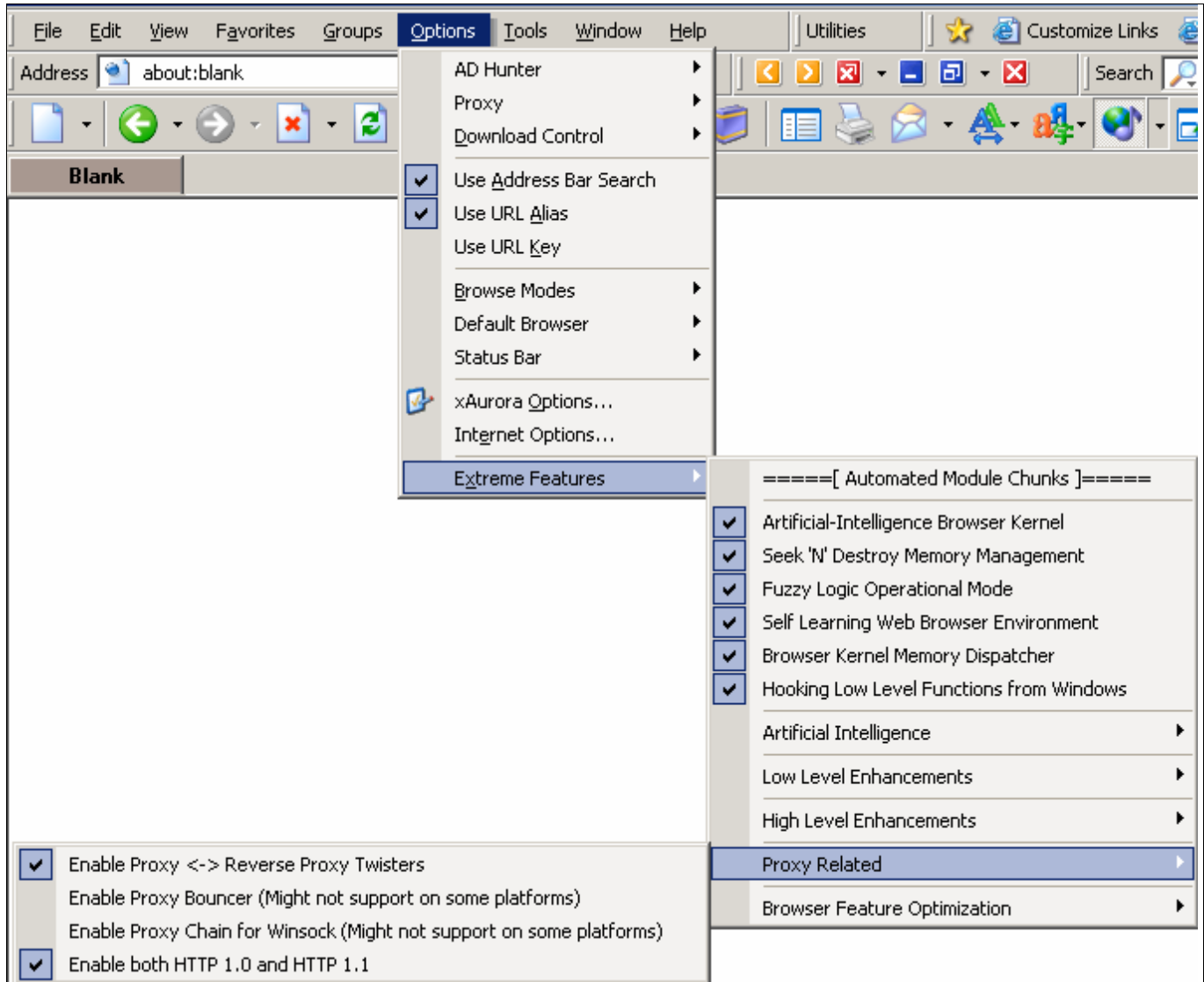
Extreme Features which Maxthon or any other browsers DOES NOT STEPPED OUT



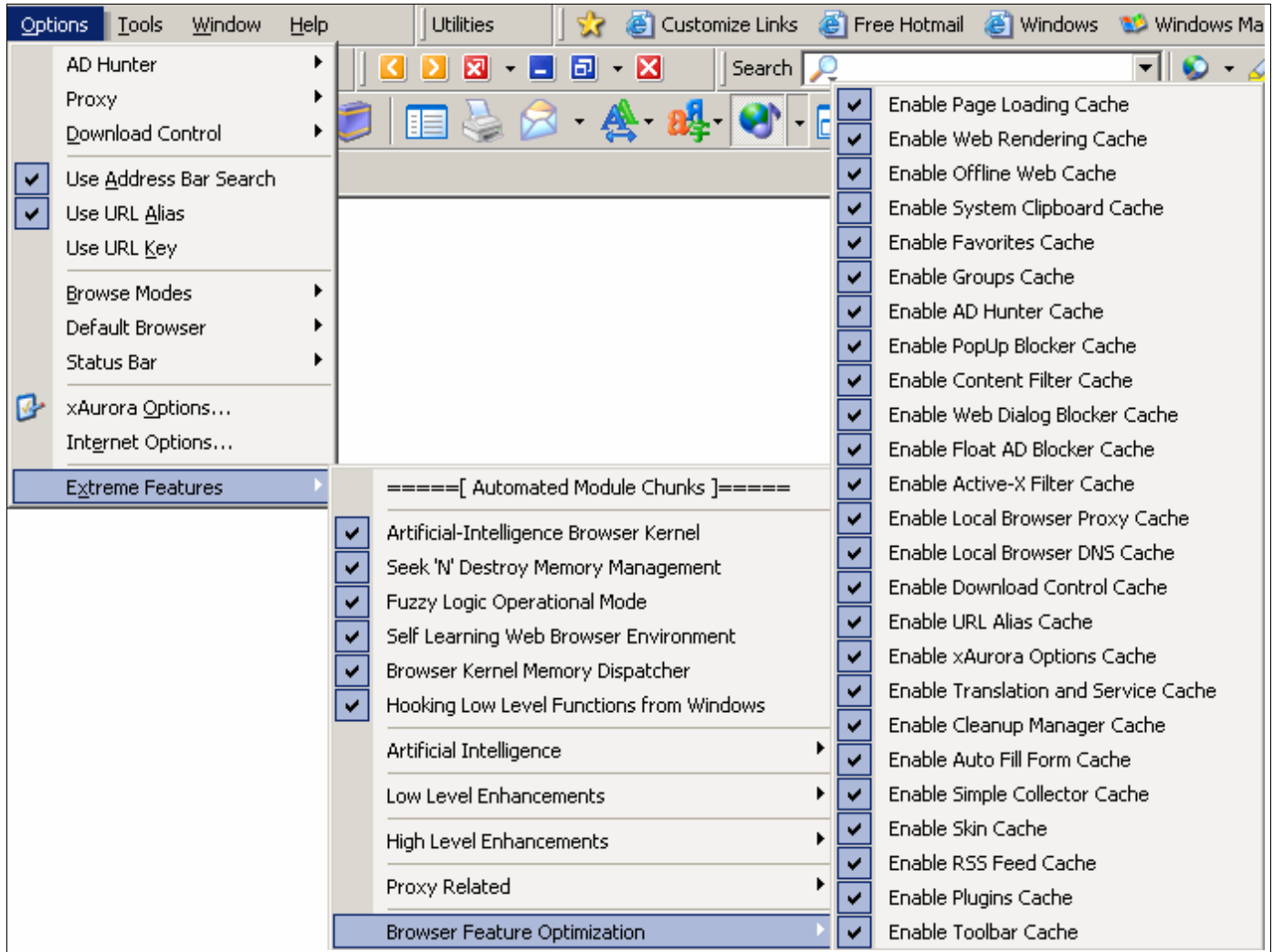
Extreme Features which Maxthon or any other browsers DOES NOT STEPPED OUT



Extreme Features which Maxthon or any other browsers DOES NOT STEPPED OUT



Extreme Features which Maxthon or any other browsers DOES NOT STEPPED OUT



Extreme Features which Maxthon or any other browsers DOES NOT STEPPED OUT

General | When Starting | When Closing

Only one instance of xAurora Enable browser hide key

Enable animated icon

Show tray icon

Minimize to system tray Show all web pages in predefine zoom factor:

Close to system tray %

Ignore window ID assignment in frames (*) Disable web script errors

Lock the Home Page Use flat browser scroll bar(*)

Add xAurora User-agent identification(*)

- Extreme features (Click only once, the variables will be written to the ini file) (+) -----


<input type="checkbox"/> Enable (AI) Artificial Intelligence engine	<input type="checkbox"/> Windows Terminal Server mode
<input type="checkbox"/> DigitalDec ALPHA and RiSC CPU mode	<input type="checkbox"/> Enable Automated CPU management
<input type="checkbox"/> Enable Anti-crash engine	<input type="checkbox"/> CPU Ring-0 framework enable
<input type="checkbox"/> Windows XP SP2 security enable	<input type="checkbox"/> Enable Self-Guard Anti-Virus module
<input type="checkbox"/> Windows 2003 SP1 security enable	<input type="checkbox"/> Intel Hyper Threading (HT) support
<input type="checkbox"/> Windows 2000 SP4 security enable	<input type="checkbox"/> Intel Itanium CPU Class I and II support
<input type="checkbox"/> Windows Vista security (Experimental)	<input type="checkbox"/> AMD Floating-Point bug fix
<input type="checkbox"/> CPU Auto-priority schedule mode	<input type="checkbox"/> AMD Athlon Acceleration Enable
<input type="checkbox"/> AMD 64 Athlon X2 and Opteron support	<input type="checkbox"/> Intel Pentium 4 Support

Default Download Control -----

<input checked="" type="checkbox"/> Load Images	<input type="checkbox"/> Share resources with IE [Internet Explorer] (+)
<input checked="" type="checkbox"/> Load Sounds	<input type="checkbox"/> Use Internet Explorer Web Active-X control (+)
<input checked="" type="checkbox"/> Load Video	<input type="checkbox"/> Own Browser Referrer (+)
<input type="checkbox"/> Play Animation	
<input checked="" type="checkbox"/> Allow Scripts	
<input checked="" type="checkbox"/> Allow Java Applet	
<input checked="" type="checkbox"/> Allow ActiveX	


Download Manager -----

Monitor File Types:

 Hold ALT key before download begins to toggle using download manager temporarily.

Extreme Features which Maxthon or any other browsers DOES NOT STEPPED OUT

General | Popup Filter | Content Filter | ActiveX Filter


- Add Ad Hunter to browser right click menu
- Enable Web AD blocker
 - Replace blocked Ad with (!)
- Filter duplicated URL
- Enable auto popup blocker
- Enable popup blocker
- Enable floating Ad blocker (!)
- Enable unwanted web dialog blocker
- Play sound when a popup window is blocked
 - ...
- Artificial Intelligence (AI) based AD hunter - Auto mode (+)
- Automated blacklisted domain filtering engine AI based (+)
- Automated malicious-code filtering AI engine (+) 

Options ended with (!) will cost more CPU power.

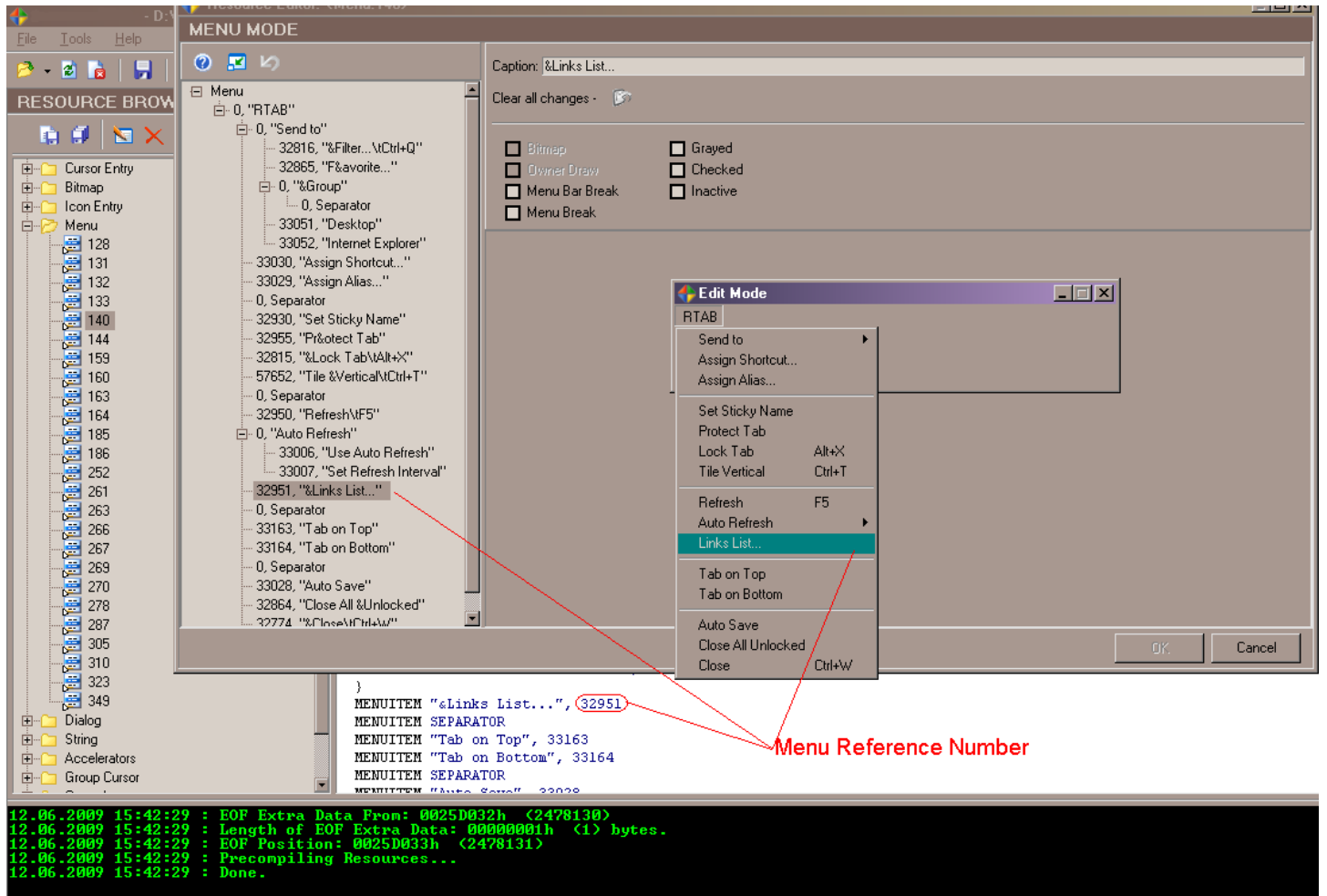
Name	Proxy	Speed
------	-------	-------

Bypass addresses beginning with: (separate by space)

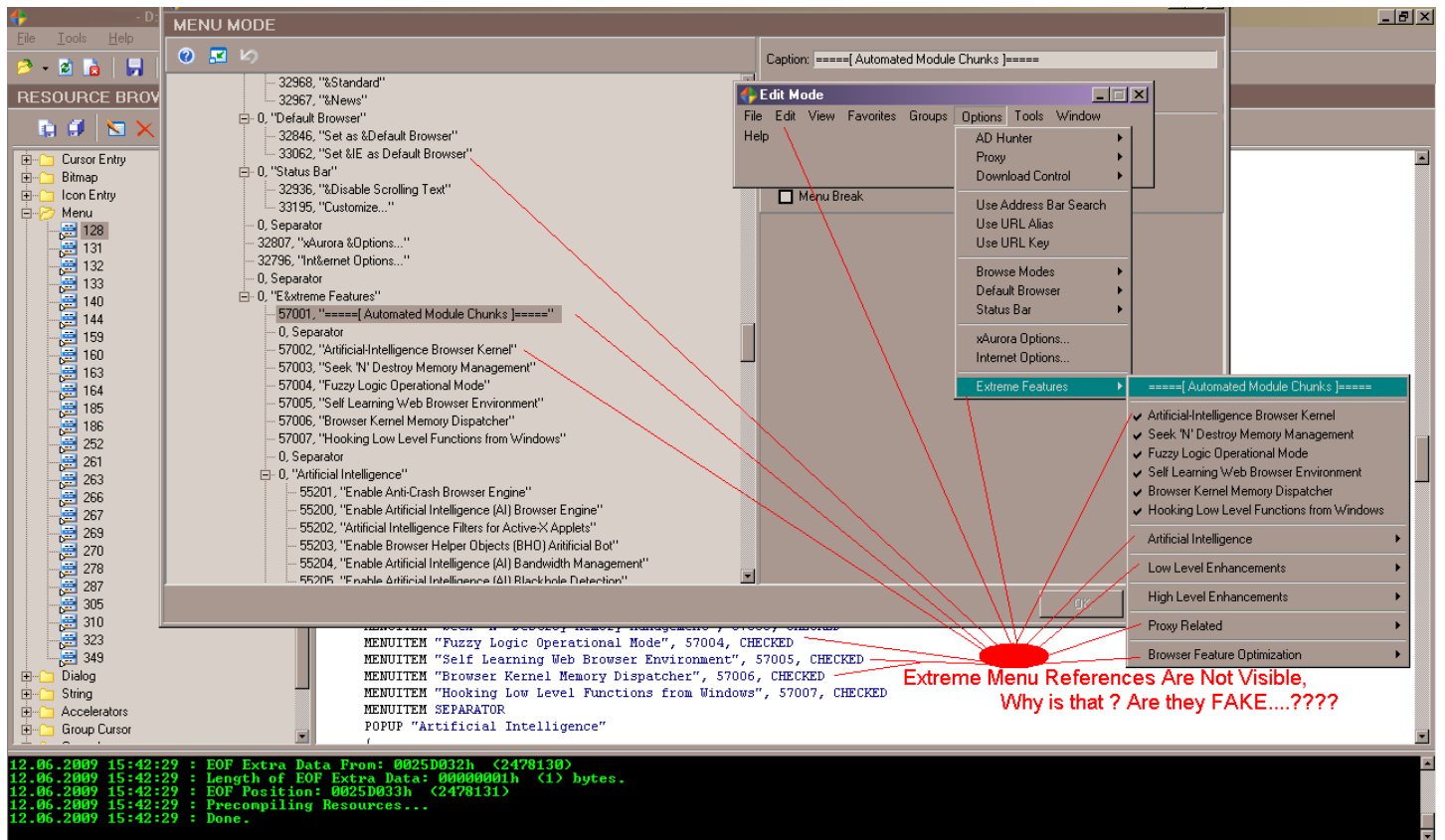
- Own Proxy and IE proxy management module (+) Reverse proxy mode (+)

 Format: IP:PORT Example: 192.168.1.10:8080
or Protocol=IP:PORT Example: http=10.1.1.0:80 socks=10.1.1.0:1080
Scope: All Windows

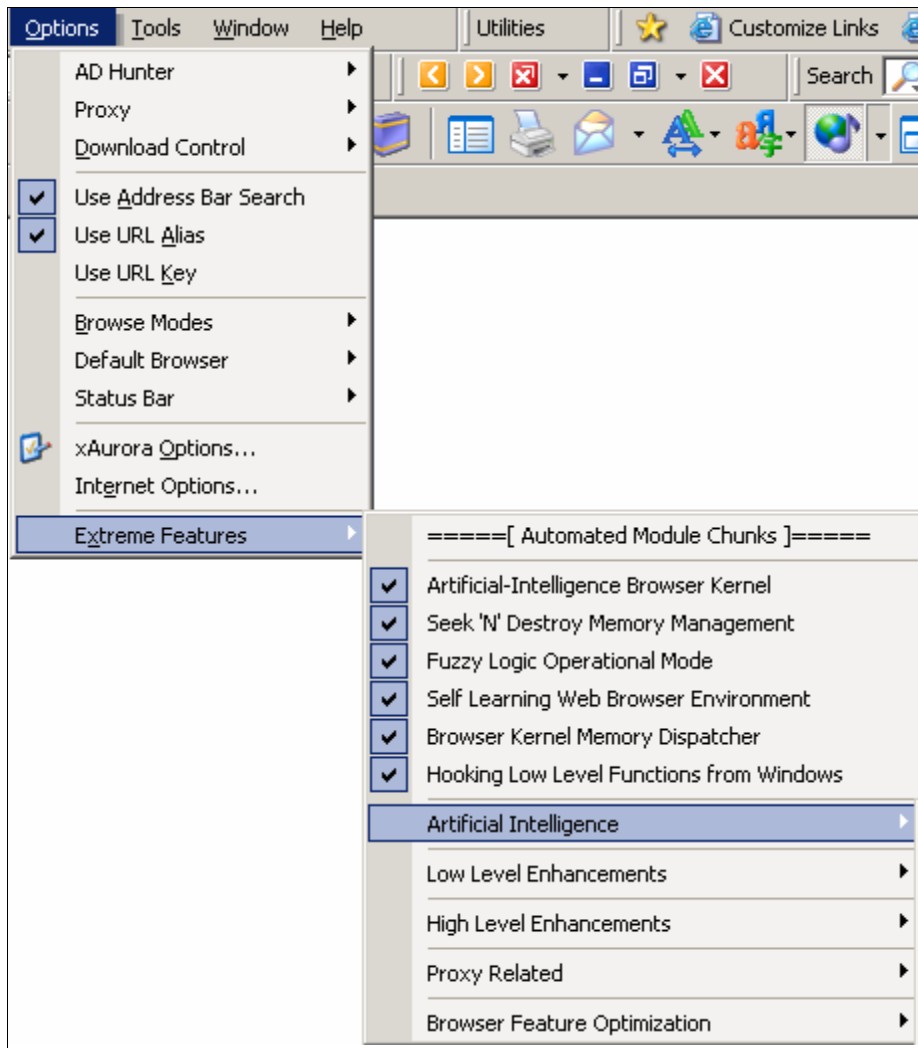
Why xAurora Extreme Features Menu References are not exposing to debugger or other low level program and the purpose?



Native Mode Menus



Special Kernel Mode Menus (Innovation from Sri Lanka)

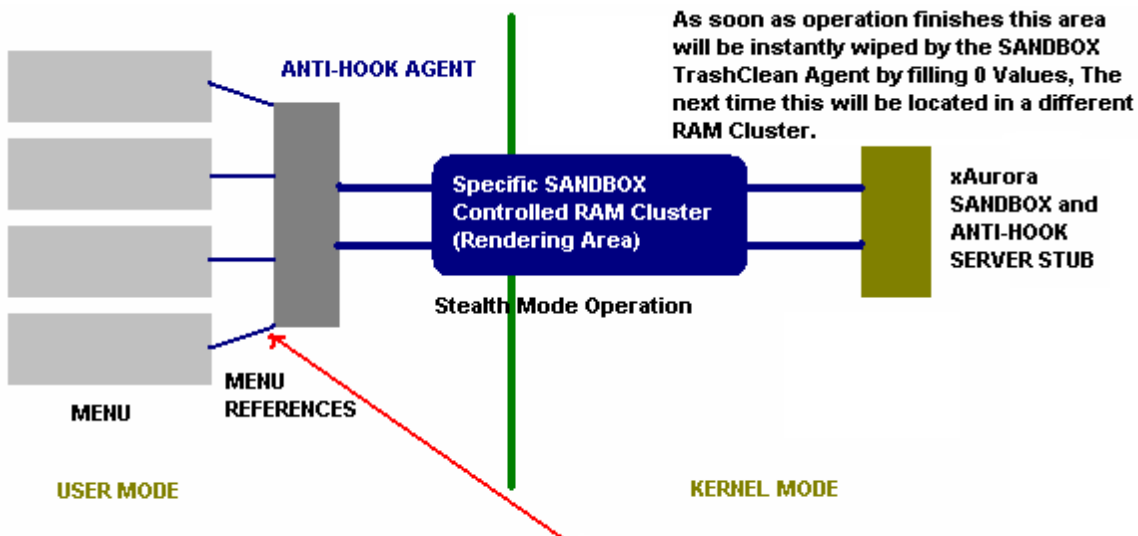


These menu references are directly connected to the Ring-0 Kernel mode operations.

If the PC already infected by Rootkit or any other Kernel mode Malware, these are the direct references to exploit and gain hooking to the Windows Kernel. Because entire set of the following menu references are working with Kernel Mode Drivers.

How we achieved it?

When the browser loads to the memory, there is a specific User Mode Anti-Hook client will be loaded with the browser. This Agent is giving the full fledge protection to the EXTREME FEATURES MENU REFERENCES Pointers.



Menu References has its own specific number "MENU REFERENCE ID". In Extreme Features in the xAurora has the different mechanism to render the Menu Reference. Menu ID will be calculated by converting MENU ITEM NAME CHARACTERS (ASCII Values) into HEX Values and then, HEX Values into OCTALS.

Ex.

MENU ITEM NAME -

CPU THREADS BOOST (CHAR-ASCII) ---->

62 e6 f1 da 4b 0c 2c 5c 27 2a 6b 26 5b 12 12 2b (HEX Values) ---->

1 427 157 073 222 606 054 270 234 523 262 313 304 411 053 (OCTAL Values) ---->

Final: 1427157073222606054270234523262313304411053 (Extreme Menu Reference ID)

ANTI-HOOK Agent Encrypts each and every Menu Reference Pointer, UID, GUID, WinCLASS, USER32.DLL-MessageBoxW(HWND, PWSTR), PID and API-Class References and Hide in the Separate Memory Area on the RAM CLUSTER in the SANDBOX MODULE. When the user click on a menu item, Agent will call SANDBOX via it's own CUSTOM API Call (SandMenuAntiHookW32) and Decrypt it on the fly and attached the MENU Reference ID and HWND Class in another SANDBOX Area. Also it will render followings, GUID will call WinCLASS APIs to gain access to the USER MODE Menu Standard Operations by calling USER32.DLL. PWSTR will be rendered as soon as the hooked each other.

END NOTE

Since I have applied the Extreme Anti Debugging Protection Schemas to xAurora Browser, Debuggers and Kernel Mode Applications will not be capturing any of the operational events of Menu References in xAurora. That's what we called "EXTREME TECHNOLOGY". The xAurora Extreme PoC (Proof of Concepts) will be finalized.

In the Kalinga's blog Mr. Prasan Hettiarchchi, Mr. Kalinga Athulathmudali, Mr. FallenZeraphine and Mr. Thathagatha tried to prove xAurora Extreme Features Menu to be Fake things. They have used Resource Viewer/Heaventools PE Explorer to harvest the core of xAurora. PE Explorer is just another general PE Editor/Resource Editor. I think they don't have much knowledge of harvesting the Deep core of a Packed/Protected executable files.

Like I have explained, in KMD(s) White Paper, I have applied many protection schemes to the browser's main executable file. Obviously, they can not prove it with using these types of LAMER TOOLS.

Also all the MENU REFERENCE KERNEL HOOKS are flowing through the Invisible Mode Anti-Hook Client to the Anti-Hook Server. Invisible KERNEL hooks will hide all the possible Kernel and Marshall (NTDLL) Hooks from xAurora and vice versa. That's why you may never been able to see any the menu reference flows running in the browser, even when you loaded browser in to any debugger. I have used most of custom made API(s) and NT Undocumented API(s) for better protection of the browser.

SPECIAL NOTE

I am sending the unwrapped (Protection Removed) xAurora Main Executable file to Mr. Gotaimbara to do the Live Demo and PoC of KMD Availability in the browser.

CONCLUSION

Kalinga's blog Mr. Prasan Hettiarchchi, Mr. Kalinga Athulathmudali, Mr. FallenZeraphine and Mr. Thathagatha's reverse engineering methodologies are very lamer and they have very basic NT Kernel/Low Level knowledge. Also they may never work with UNDOCUMENTED NT API(s) in NT Subsystem.

This guideline document will help you to understand why Mr. Prasan Hettiarchchi, Mr. Kalinga Athulathmudali, Mr. FallenZeraphine and Mr. Thathagatha's xAurora Kernel Mode Extreme Features Menu References PoC failed and they couldn't prove it.

I know my own code better than all of you. Because, I am the founder of xAurora's concepts and I am the programmer of the xAurora Web Browser. No one can admit the wrong conclusion without proving it in the real world and to the community. Because, xAurora is a COPYRIGHTED, TRADEMARKED and PATENTED SOFTWARE.

xAurora is a great Sri Lankan product that entirely coded in Win32 Assembly Language. Hope you may be able to understand it. In future I will show you many stories behind the xAurora case. Hope Mr. Gotaimbara will help me to sort out the matters soon. Thank you very much for the great support and your support in the future is greatly appreciated.

Kind Regards

Dr. Sameera de Alwis

Founder – Team xAurora 2009