

Forum:[Hack Forums](#)

Thread: [OffCode | Web Remote Control \[FREE \]](#)

OffCode - May 4, 2026, 1:00 pm

The Web Remote Control System is a high-performance, completely free, and self-hosted platform that lets you manage Windows devices directly from any web browser. Combining a lightweight Windows Agent with a Blazor-based control panel, you can monitor and control remote computers in real time without being physically present.

AES-256 GCM Encryption: The system is built with security as its absolute foundation. All sensitive payloads and communication between the Agent and the Backend are encrypted end-to-end using AES-256 GCM. This guarantees that even if your connection is intercepted using packet sniffers like Wireshark, the data remains completely unreadable and protected. A dedicated Server Manager tool securely generates cryptographic Master Keys and directly embeds them into the Agent executables, ensuring zero-configuration deployment without compromising security.

Video:

https://youtu.be/_gMLGK7DUUI

? Features

AES-256 GCM Encryption: Sensitive hub payloads are encrypted at the application layer; use HTTPS/WSS in production. Each agent stub gets its own AES-256 key (AgentKeyId) plus a per-stub connect token.

Live Stream & Interactive Control: View the remote desktop and interact with the mouse/keyboard in real time.

Chat: Real-time two-way chat between the dashboard administrator and the remote agent user, with a customizable admin nickname and message history.

Share Screen (Public Viewer): Generate a public, anonymous link to securely share the live stream without giving dashboard access. Custom expiry (hours + minutes) or No time limit; expired links stop relaying frames automatically.

Streaming File Manager: Transfer large files efficiently without hitting memory limits; browse all ready drives (fixed, removable, network) and jump to common folders (e.g. Desktop, Documents, Downloads, Program Files, AppData); click-to-open navigation with upload target following the current path; optional Upload and run toggle; File Search by name with wildcard and recursive subdirectory support.

Desktop Screenshot: Capture preview in the control panel; Close Image dismisses the preview without leaving the page.

Task Manager & Process Blocker: View and manage active processes from the web interface. Block specific applications by name so the remote agent kills them on startup; setting persists even if the dashboard is closed.

Power Controls: Shut down or restart the remote device from the Task Manager panel.

Software Manager: List installed applications and uninstall them remotely using Windows Registry data.

Remote Process Execution: Instantly run commands, open files, or launch software on the remote machine.

Remote Terminal & Script Manager: Interactive PowerShell/CMD terminal in the browser, plus a built-in script editor for Python, PowerShell, Batch, and VBScript. Includes Stop Script, Download Output, and Scheduled Script Execution (managed by the backend even if the browser dashboard is closed).

System Telemetry: Monitor hardware and software metrics (CPU, RAM, OS version, network info) through the Computer Info dashboard. Startup Programs includes a Remove button (Registry HKCU/HKLM or startup folder; admin required for HKLM).

Device History: Track and manage previously connected devices for quicker reconnection.

Real-time Device Ping: Connected devices list shows live round-trip latency (ms) between the backend and each online agent, updated about every 10 seconds with color-coded indicators.

Telegram Notifications (Optional): Receive a Telegram message when any device comes online. Configure bot token and chat ID from Agent settings; includes setup help and a test message button.

Telegram Bot Commands: Send /Get_info in the configured Telegram chat to fetch Computer Info from an online device (device selection when multiple devices are connected).

Agent Settings: Windows auto startup with a custom startup name (default: RemoteControlAgent), configurable from the dashboard.

Dashboard Lock: Password gate for the admin panel (/authentication/login), with settings at /authentication/settings. Cookie session plus short-lived JWT for SignalR and protected REST APIs.

Remove Agent: Disconnect a device from the dashboard and trigger remote uninstall of the agent executable on the target PC; if the agent is offline, the request is queued in SQLite and runs when it reconnects.

Server Manager: Configure ports, start the backend and frontend independently (stopping Frontend does not stop Backend, and vice versa), generate secure client agents (stubs), toggle optional file logging, and view the latest project changelog directly within the application.

Download: <https://www.dev-offcode.com/RemoteControl.html>

DramaKing - May 4, 2026, 1:45 pm

So it's free except for the source code?

OffCode - May 4, 2026, 1:53 pm

(May 4, 2026 01:45 PM)DramaKing Wrote: [\[CITE\]](#) So it's free except for the source code?
Yes! I will upload more pictures tomorrow!

5chn4pp - May 4, 2026, 1:58 pm

It's based on the Blazor framework? That's hilarious.

Since you embed keys into the agent, is it mTLS based? I may be reading that wrong.

OffCode - May 4, 2026, 2:02 pm

(May 4, 2026 01:58 PM)5chn4pp Wrote: [\[CITE\]](#) It's based on the Blazor framework? That's hilarious.

Since you embed keys into the agent, is it mTLS based? I may be reading that wrong.

Yes, the dashboard uses Blazor Server because its seamless integration with SignalR makes real-time, bi-directional communication incredibly smooth when sharing C# models across the stack.

Regarding the keys: It's actually not mTLS. The system uses Application-Layer Payload Encryption. The embedded keys inside the agent consist of an API Key (used to authenticate and fetch a JWT Bearer token) and a shared AES-256 GCM key.

The actual payload data (commands, telemetry, inputs) is encrypted at the application layer using AES-256 before being sent over SignalR. The transport layer just relies on standard HTTPS/WSS via a reverse proxy (like Nginx), but even if the TLS is terminated at the proxy, the payloads remain completely unreadable without the AES key.

5chn4pp - May 4, 2026, 2:12 pm

(May 4, 2026 02:02 PM)M A G U Wrote: [\[CITE\]](#) Yes, the dashboard uses Blazor Server because its seamless integration with SignalR makes real-time, bi-directional communication incredibly smooth when sharing C# models across the stack.

Regarding the keys: It's actually not mTLS. The system uses Application-Layer Payload Encryption. The embedded keys inside the agent consist of an API Key (used to authenticate and fetch a JWT Bearer token) and a shared AES-256 GCM key.

The actual payload data (commands, telemetry, inputs) is encrypted at the application layer using AES-256 before being sent over SignalR. The transport layer just relies on standard HTTPS/WSS via a reverse proxy (like Nginx), but even if the TLS is terminated at the proxy, the payloads remain completely unreadable without the AES key.

Sounds decent, as long as the agent key isn't the same for everyone.

Otherwise one key could decrypt everyone else's traffic. :)

OffCode - May 4, 2026, 2:19 pm

(May 4, 2026 02:12 PM)5chn4pp Wrote: [\[CITE\]](#) Sounds decent, as long as the agent key isn't the same for everyone.

Otherwise one key could decrypt everyone else's traffic. :)Yeah!

The AES-256 GCM key is completely unique per installation. When the user opens the Server Manager for the first time, the system generates unique key

I might actually add a button in the future to let the user manually regenerate and change the key themselves too!

jason123 - May 4, 2026, 3:42 pm

(May 4, 2026 01:00 PM)M A G U Wrote: [\[CITE\]](#) The Web Remote Control System is a high-performance, completely free, and self-hosted platform that lets you manage Windows devices directly from any web browser. Combining a lightweight Windows Agent with a Blazor-based control panel, you can monitor and control remote computers in real time without being physically present.

AES-256 GCM Encryption: The system is built with security as its absolute foundation. All sensitive payloads and communication between the Agent and the Backend are encrypted end-to-end using AES-256 GCM. This guarantees that even if your connection is intercepted using packet sniffers like Wireshark, the data remains completely unreadable and protected. A dedicated Server Manager tool securely generates cryptographic Master Keys and directly embeds them into the Agent executables, ensuring zero-configuration deployment without compromising security.

? Features

AES-256 GCM Encryption: All sensitive data is encrypted end-to-end.

Live Stream & Interactive Control: View the remote desktop and interact with the mouse/keyboard in real time.

Streaming File Manager: Transfer large files efficiently without hitting memory limits.

Task Manager: View active processes and manage them directly from the web interface.

Remote Process Execution: Instantly run commands, open files, or launch software on the remote machine.

System Telemetry: Monitor detailed hardware and software metrics (CPU, RAM, OS version) through the Computer Info dashboard.

Server Manager: A built-in tool to easily configure ports, launch the server, and generate secure client agents (stubs).

Download: <https://www.dev-offcode.com/RemoteControl.html>

Hey MAGU, thanks for this software, it worth testing to see the effect. Does it need to be crypted or it bypasses WD?

ServersFxcked - May 4, 2026, 9:13 pm

(May 4, 2026 03:42 PM)jason123 Wrote: [\[CITE\]](#) Hey MAGU, thanks for this software, it worth testing to see the effect. Does it need to be crypted or it bypasses WD?

Really make a output and go test it, brother doesnt have to wipe your ass facepalm*

OffCode - May 5, 2026, 1:10 am

(May 4, 2026 03:42 PM)jason123 Wrote: [\[CITE\]](#) Hey MAGU, thanks for this software, it worth testing

to see the effect. Does it need to be crypted or it bypasses WD?

It doesn't need to be an crypted stub at the moment. As time goes on and downloads increase, antivirus software may start flagging it anyway, even if the program itself doesn't contain any particularly dangerous or harmful features.

Added more pictures to our website

jason123 - May 5, 2026, 3:55 am

(May 4, 2026 09:13 PM)ServersFxcked Wrote: [\[CITE\]](#) Really make a output and go test it, brother doesnt have to wipe your ass facepalm*

You shoud not have responded to what you do not know much about. There is no harm in asking question! Fair enough OP responded gently

(May 5, 2026 01:10 AM)M A G U Wrote: [\[CITE\]](#) It doesn't need to be an crypted stub at the moment. As time goes on and downloads increase, antivirus software may start flagging it anyway, even if the program itself doesn't contain any particularly dangerous or harmful features.

Added more pictures to our website

Noted with thanks ?.

ServersFxcked - May 5, 2026, 4:04 am

[quote="jason123" pid='62950902' dateline='1777978522']

You shoud not have responded to what you do not know much about. There is no harm in asking question! Fair enough OP responded gently

I sell my Own Rat and Crypter so I do but algoood, I waas just making a point saying you could've done it yourself instead of waiting for a response

ColdBoot - May 5, 2026, 2:14 pm

Program doesn't work and requires .net sdk which isn't mentioned in the readme

OffCode - May 6, 2026, 4:30 am

(May 5, 2026 02:14 PM)ColdBoot Wrote: [\[CITE\]](#) Program doesn't work and requires .net sdk which isn't mentioned in the readme

Program works fine, Yes its requires .net SDK i write it to readme.

You did portforwarding?

ColdBoot - May 6, 2026, 10:57 am

(May 6, 2026 04:30 AM)M A G U Wrote: [\[CITE\]](#) Program works fine, Yes its requires .net SDK i write it to readme.

You did portforwarding?

Has nothing to do with port forwarding. The services for front and backend can't even start.

Code [13:53:23] Log cleared by user

[13:53:25] [Frontend Only] Starting frontend server...

[13:53:25] AES_MASTER_KEY_HEX not found in appsettings or .env; key will not be written to .env.

[13:53:25] Frontend configuration synced (appsettings.json + .env).

[13:53:25] launchSettings.json not found at FrontEnd\Properties\launchSettings.json

[13:53:25] [Frontend Only] Frontend started successfully (Published DLL mode)
[13:53:26] [Frontend Error] Could not execute because the specified command or file was not found.
[13:53:26] [Frontend] Possible reasons for this include:
[13:53:26] [Frontend] * You misspelled a built-in dotnet command.
[13:53:26] [Frontend] * You intended to execute a .NET program, but dotnet-FrontEnd\FrontEnd.dll does not exist.
[13:53:26] [Frontend] * You intended to run a global tool, but a dotnet-prefixed executable with this name could not be found on the PATH.
[13:53:26] [Frontend Only] Frontend process exited unexpectedly
By the way just running the manager as extracted doesn't even work, I have to move the other folders into Manager folder just so it doesn't crash.

Code See the end of this message for details on invoking just-in-time (JIT) debugging instead of this dialog box.

***** Exception Text *****

System.InvalidOperationException: No process is associated with this object.
at System.Diagnostics.Process.EnsureState(State state)
at System.Diagnostics.Process.get_HasExited()
at ServerManager.Form1.UpdateUI() in C:\Users\35844\Desktop\RemoteControl Open\ServerManager\Form1.cs:line 1091
at ServerManager.Form1.btnStart_Click(Object sender, EventArgs e) in C:\Users\35844\Desktop\RemoteControl Open\ServerManager\Form1.cs:line 250
at System.Threading.Tasks.Task.<>c.<ThrowAsync>b__128_0(Object state)
at System.Windows.Forms.Control.InvokeMarshaledCallbackHelper(Object obj)
at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext, ContextCallback callback, Object state)
--- End of stack trace from previous location ---
at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext, ContextCallback callback, Object state)
at System.Windows.Forms.Control.InvokeMarshaledCallbacks()

OffCode - May 6, 2026, 11:15 am

(May 6, 2026 10:57 AM)ColdBoot Wrote: [\[CITE\]](#) Has nothing to do with port forwarding. The services for front and backend can't even start.

Code [13:53:23] Log cleared by user
[13:53:25] [Frontend Only] Starting frontend server...
[13:53:25] AES_MASTER_KEY_HEX not found in appsettings or .env; key will not be written to .env.
[13:53:25] Frontend configuration synced (appsettings.json + .env).
[13:53:25] launchSettings.json not found at FrontEnd\Properties\launchSettings.json
[13:53:25] [Frontend Only] Frontend started successfully (Published DLL mode)
[13:53:26] [Frontend Error] Could not execute because the specified command or file was not found.
[13:53:26] [Frontend] Possible reasons for this include:
[13:53:26] [Frontend] * You misspelled a built-in dotnet command.
[13:53:26] [Frontend] * You intended to execute a .NET program, but dotnet-FrontEnd\FrontEnd.dll does not exist.
[13:53:26] [Frontend] * You intended to run a global tool, but a dotnet-prefixed executable with this name could not be found on the PATH.
[13:53:26] [Frontend Only] Frontend process exited unexpectedly

By the way just running the manager as extracted doesn't even work, I have to move the other folders into Manager folder just so it doesn't crash.

Code See the end of this message for details on invoking just-in-time (JIT) debugging instead of this dialog box.

***** Exception Text *****

```
System.InvalidOperationException: No process is associated with this object.
  at System.Diagnostics.Process.EnsureState(State state)
  at System.Diagnostics.Process.get_HasExited()
  at ServerManager.Form1.UpdateUI() in C:\Users\35844\Desktop\RemoteControl
Open\ServerManager\Form1.cs:line 1091
  at ServerManager.Form1.btnStart_Click(Object sender, EventArgs e) in
C:\Users\35844\Desktop\RemoteControl Open\ServerManager\Form1.cs:line 250
  at System.Threading.Tasks.Task.<>c.<ThrowAsync>b__128_0(Object state)
  at System.Windows.Forms.Control.InvokeMarshaledCallbackHelper(Object obj)
  at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext,
ContextCallback callback, Object state)
--- End of stack trace from previous location ---
  at System.Threading.ExecutionContext.RunInternal(ExecutionContext executionContext,
ContextCallback callback, Object state)
  at System.Windows.Forms.Control.InvokeMarshaledCallbacks()
1.Start server manager
2.Setup server address
3.Frontend port
4.Backend port
5.Go settings page
6.Press save
7.Try again.
```

Do you see secret key when you start server manager?

ColdBoot - May 6, 2026, 12:26 pm

Secret key is blank

Try running the program you posted on github in a clean VM environment.

Pyr00 - May 6, 2026, 7:43 pm

It's possible I misunderstood your reply earlier but is it a single AES key for every connection to share? Or is each individual connection encrypted with a different key?

The former would be a pretty serious design flaw in terms of security. Which one is it?

Cool project though, are you not releasing the source because you are interested in commercializing down the line?

If you did release it, it would become a lot more popular with more than just skids as we can audit the code for ourselves.

But anyways, good luck with future development!

DeathPenalty666 - May 6, 2026, 7:51 pm

(May 6, 2026 07:43 PM)Pyr00 Wrote: [\[CITE\]](#) It's possible I misunderstood your reply earlier but is it a single AES key for every connection to share? Or is each individual connection encrypted with a

different key?

The former would be a pretty serious design flaw in terms of security. Which one is it?

Cool project though, are you not releasing the source because you are interested in commercializing down the line?

If you did release it, it would become a lot more popular with more than just skids as we can audit the code for ourselves.

But anyways, good luck with future development!

The AES-256 GCM key is completely unique per installation. When the user opens the Server Manager for the first time, the system generates unique key

each user have a unique key as i can read here.

OffCode - May 7, 2026, 12:06 am

(May 6, 2026 12:26 PM)ColdBoot Wrote: [\[CITE\]](#) Secret key is blank

Try running the program you posted on github in a clean VM environment.

Ok, I try make little fix. Thanks for logs i think i found some problems.

OffCode - May 8, 2026, 12:38 pm

New version is available now!

Fixed crashing and empty secret key problem.

(May 6, 2026 07:43 PM)Pyr00 Wrote: [\[CITE\]](#) It's possible I misunderstood your reply earlier but is it a single AES key for every connection to share? Or is each individual connection encrypted with a different key?

The former would be a pretty serious design flaw in terms of security. Which one is it?

Cool project though, are you not releasing the source because you are interested in commercializing down the line?

If you did release it, it would become a lot more popular with more than just skids as we can audit the code for ourselves.

But anyways, good luck with future development!

Thank you very much!

Every user have unique key

I'm not sure yet whether I'll release the source code for free or sell it. For now, it will be for free without the source code. I do understand that releasing the source code as well would improve transparency and trustworthiness.

OffCode - May 11, 2026, 9:31 am

NEW UPDATE: 09/05/2026

Added: Software Manager — Remotely list and uninstall applications via Windows Registry.

Added: Power Controls — Shutdown and Restart devices remotely from the Task Manager panel.

Added: Device History — Track and manage previously connected devices.

Added: File Manager — dynamic drives — Quick-access buttons for all ready drives (fixed, removable, network) fetched from the agent over the encrypted SignalR channel.

Added: File Manager — quick folders — Program Files, Program Files (x86), and Roaming AppData shortcuts (resolved on the agent via shell folder tokens, same pattern as Desktop/Documents/Downloads).

Improved: File Manager UX — Open folders with a single click on the row; upload target automatically follows the current browse path (no separate “Set as Target” step).

Added: File Manager — upload and run — Optional checkbox to open uploaded files automatically on the agent after transfer completes.

Added: Desktop Screenshot — Close Image — Dismiss the screenshot preview from the device control UI.

Added: Server Manager — log toggle — Checkbox to turn application logging on or off.

Fixed: Server Manager — Stopping or restarting the frontend no longer stops the backend (independent service controls).

Added: Dashboard Lock (optional) — Cookie session for the Blazor UI, lock-aware redirects, /authentication/continue when lock is disabled, safe returnUrl handling; backend stores the dashboard password as a secure hash; short-lived JWT for admin SignalR and protected REST (DeviceController, admin FileManager routes); agent HTTP file endpoints (agent-download / agent-upload) remain reachable without JWT; configure requires auth or DASHBOARD_LOCK_ADMIN_SECRET.

Added: Remove Agent — Remote uninstall flow with confirmation; offline queue in SQLite (PendingAgentCommands) until the agent reconnects.

Added: Lock settings UI — Dashboard button on /authentication/settings opens the devices page (/devices).

OffCode - May 13, 2026, 5:49 pm

UPDATE AVAILABLE!

14/05/2026

VER: 0.3

Added: Share Screen - Share the agent's live stream with anyone via a secure, public link without requiring a dashboard login. The public viewer connects anonymously while your dashboard acts as the decryption relay.

Added: Remote Terminal - Fully interactive PowerShell and Command Prompt terminal directly in the web dashboard.

Added: Script Manager - Built-in code editor to write, save, and execute scripts (Python, PowerShell, Batch, VBScript) on remote agent machines. Scripts are safely stored in browser's local storage for easy reuse.

OffCode - May 18, 2026, 6:52 am

NEW UPDATE IS AVAILABLE!

18/05/2026

VER: 0.4

Changes:

Added: Script manager tab / Scheduled Script Execution — which allows setting a specific date and time for a script to run automatically, independently managed by the backend server.

Added: Server Manager Changelog — In-app changelog viewer that fetches and parses the latest release notes automatically, along with a quick link to download new versions.

DeathPenalty666 - May 18, 2026, 7:03 am

(May 18, 2026 06:52 AM)M A G U Wrote: [\[CITE\]](#) NEW UPDATE IS AVAILABLE!

18/05/2026

VER: 0.4

Changes:

Added: Script manager tab / Scheduled Script Execution — which allows setting a specific date and time for a script to run automatically, independently managed by the backend server.

Added: Server Manager Changelog — In-app changelog viewer that fetches and parses the latest release notes automatically, along with a quick link to download new versions.

damn i left you away a couple days and you made it a realllllllllly interesting projet!! i gotta get the source :) you have any eta on a date?
and you will have any range of price for the sorce on automation too?

OffCode - May 18, 2026, 7:46 am

(May 18, 2026 07:03 AM)DeathPenalty666 Wrote: [\[CITE\]](#) damn i left you away a couple days and you made it a realllllllllly interesting projet!! i gotta get the source :) you have any eta on a date?

and you will have any range of price for the sorce on automation too?

The source code will likely cost around \$200. The source code is not yet available for sale, but I can let you know when it becomes available for purchase.

DeathPenalty666 - May 18, 2026, 8:24 am

(May 18, 2026 07:46 AM)M A G U Wrote: [\[CITE\]](#) The source code will likely cost around \$200. The source code is not yet available for sale, but I can let you know when it becomes available for purchase.

When you have any hind au automation and a date for this just ping me up it!!!! i cant wish any better than a hvnc verion but i totally want this on my project :D

automation will just be a extra layer of advantager :)

OffCode - May 18, 2026, 8:40 am

(May 18, 2026 08:24 AM)DeathPenalty666 Wrote: [\[CITE\]](#) When you have any hind au automation and a date for this just ping me up it!!!! i cant wish any better than a hvnc verion but i totally want this on my project :D

automation will just be a extra layer of advantager :)

Did you mean that you want automation features for the web remote control system, or were you asking about the price of my automation tool source code?

DeathPenalty666 - May 18, 2026, 8:47 am

(May 18, 2026 08:40 AM)M A G U Wrote: [\[CITE\]](#) Did you mean that you want automation features for the web remote control system, or were you asking about the price of my automation tool source code?

i wish the both project in mine :D So anytime you have need for any on those hit me up. automation multi os would be a dream but i can live with iut on a simge mahcine for some time. i basically will use it ONE DEVICE to automate a bunch of my device for various remote control task. BUT IF YOU MEAN There is hope to have automation literally integrated into this now i dont want it ill neeeeed it.

OffCode - May 18, 2026, 8:51 am

(May 18, 2026 08:47 AM)DeathPenalty666 Wrote: [\[CITE\]](#) i wish the both project in mine :D So anytime you have need for any on those hit me up. automation multi os would be a dream but i can live with iut on a simge mahcine for some time. i basically will use it ONE DEVICE to automate a bunch of my device for various remote control task. BUT IF YOU MEAN There is hope to have automation literally integrated into this now i dont want it ill neeeeed it.

Okay! I will contact you soon!

Tomorrow i will update pictures to website

OffCode - May 27, 2026, 10:39 am

I'm always open to feature suggestions. In the next update, I'm planning to improve the live remote control feature.

OffCode - June 4, 2026, 6:14 am

Showcase video: https://youtu.be/_gMLGK7DUUI

OffCode - June 13, 2026, 7:08 am

UPDATE AVAILABLE!

13/06/2026

VER: 0.5

Added: Process Blocker — Prevent specific applications from running on the remote device. Enter the process name in the Task Manager section, and the background agent service will instantly kill

it whenever it attempts to start. The blocking setting is saved directly to the Windows Registry and persists across reboots and network reconnects.

Added: Startup Programs — Remove — Each startup program entry in Computer Info now has a Remove button. Clicking it sends an encrypted RequestRemoveStartupProgram command through the hub to the agent, which deletes the registry key (HKCU\.\Run, HKLM\.\Run, or Wow6432Node) or the startup folder shortcut. The result is returned over the same encrypted channel and displayed as a toast notification (success / error). HKLM removal requires the agent to be running as Administrator.

Added: File Manager — Search — Search for files by name in the current directory (including subdirectories) directly from the web dashboard. Results are capped at 500 items for safety and use the encrypted SignalR tunnel for communication.

OffCode - June 18, 2026, 7:57 am

NEW UPDATE! 18/06/2026

VER: 0.6

ChangesSecurity: Request Timeout / Slowloris Protection — RequestTimeout middleware added (30 s global, 10 min for file transfers, ? for SignalR). Kestrel hardened with RequestHeadersTimeout: 15 s and KeepAliveTimeout: 120 s. HTTP/2 stream limit set to 100 per connection.

Security: Error Message Hardening — Internal ex.Message no longer exposed in HTTP responses across UserController, DeviceController, and FileManagerController (12 locations fixed). All 500 responses now return a safe generic message; full exception details logged server-side only. Global fallback exception handler added to BackEnd/Program.cs with correlationId for log tracing.

Fixed: Script Execution Timeout Removed — Scripts now run indefinitely until manually stopped by the user. The previous 5-minute hard timeout (CancellationTokenSource(TimeSpan.FromMinutes(5))) has been removed from ScriptExecutionService. Output buffer size limit also removed.

Improved: UI Modernization (Glassmorphism) — The Blazor FrontEnd has been significantly upgraded with a sleek “glassmorphism” theme featuring emerald/teal accents. This applies to LiveStream, DeviceControl, AgentSettingsPanel, and authentication pages.

Fixed: Mobile Layout — Connected devices and history views are now fully responsive on mobile devices via a robust Flexbox layout, fixing overflowing text and stretched elements.

Improved: Live Stream Viewer — Added dynamic zoom controls allowing users to zoom in/out with the mouse wheel while tracking the cursor position. Added support for up to 240 FPS and dynamic quality settings (10–100%) for incredibly smooth playback, with 30 FPS set as the new default.

Fixed: Share Screen Link Copy — The “Copy Link” button in the Live Stream view now uses synchronous JavaScript logic for improved reliability across all environments.

Added: Stealth Mode — You can now hide the agent's tray icon directly from the "Agent settings" panel. When Stealth Mode is enabled, the agent runs silently in the background and is only visible in the Windows Task Manager.

Added: Chat — Real-time two-way chat window between the dashboard administrator and the remote agent's user, complete with customizable administrator nickname and message history.

Added: Script Manager Enhancements (Job-based) — The Remote Agent now tracks all running and completed scripts directly in its RAM. The dashboard automatically fetches this real-time status, ensuring you never lose track of a script's progress, even if you close the browser. Includes a new "Clear Recent" functionality, safe asynchronous script stopping, and the ability to download execution output.

OffCode - June 20, 2026, 10:27 am

UPDATE AVAILABLE!

VER: 0.7

20/06/2026

Changes

Added: Reverse Proxy Support — Built-in support for Nginx and Cloudflare Tunnels utilizing X-Forwarded-For and X-Forwarded-Proto headers. The servers now seamlessly track real client IP addresses securely when placed behind a reverse proxy.

Added: Auto Startup Name modal — Enabling Windows auto startup from Agent settings now opens a modal where you can choose a custom startup name (or keep default RemoteControlAgent).

Improved: Live Stream Cursor Visibility — The real system cursor from the agent machine is now rendered directly into captured livestream frames, including scaled captures.

Improved: Live Stream Smoothness — Streaming lag/stutter has been significantly reduced with pipeline optimizations, resulting in much smoother playback and fewer frame hiccups.

Fixed: Remote Control Right Click — Right-click from dashboard now correctly executes as right-click on the agent side, and right mouse button no longer triggers left-button down/up drag events.

OffCode - July 5, 2026, 7:02 am

05/07/2026

UPDATE AVAILABLE!

VER: 0.8

Changes

Security: Agent encryption & identity (v2) — Server Manager now generates a unique AES-256 key, AgentKeyId, and AGENT_CONNECT_TOKEN (256-bit random hex) for each new stub. All three are embedded in the stub trailer/.env and registered in BackEnd/AgentKeys.json (hot-reloaded via AgentKeyStore, no BackEnd restart). Per-agent keys limit blast radius: leaking one stub no longer decrypts every agent's traffic (the shared AES_MASTER_KEY_HEX remains BackEnd<->FrontEnd dashboard only). Connect token closes identity spoofing on the anonymous ConnectAsDevice hub method: the BackEnd validates the token with constant-time hex comparison before registration; new agents call ConnectAsDevice directly over TLS. Validation activates per registry entry (legacy rows without connectToken still work until updated). The encrypted ReceiveEncryptedFromAgent -> ConnectAsDevice bypass is restricted when AgentSecurity:RequireConnectToken is true (false in dev, true in appsettings.Production.json). Production: rebuild all agent stubs after upgrading the BackEnd, then enable RequireConnectToken=true. Use HTTPS/WSS so the connect token is not visible on the wire. LoadTester supports direct token-based registration.

Fixed: FrontEnd Hardcoded Dev Fallback Key Removed — The FrontEnd's encryption service previously fell back to a hardcoded AES key when ASPNETCORE_ENVIRONMENT=Development and AES_MASTER_KEY_HEX was unset. It now fails fast and refuses to start without a valid key, matching the BackEnd and RemoteAgent behavior exactly.

MadScientist - July 8, 2026, 2:33 pm

Anyone have an actual review? Curious..

OffCode - July 9, 2026, 3:27 am

(Jul 8, 2026 02:33 PM)MadScientist Wrote: [\[CITE\]](#) Anyone have an actual review? Curious..

Not yet, unfortunately. I'd really appreciate some honest feedback. I've done a lot of testing myself and spent a lot of time hunting for bugs and making the software as easy to use as possible, but feedback from other people is always valuable.

OffCode - July 11, 2026, 12:00 am

NEW UPDATE: 10/07/2026

VER: 0.9

Changes:

Security: ConnectAsDevice bypass fix — Fixed a security vulnerability in DeviceControlHub.ConnectAsDevice. When AgentSecurity:RequireConnectToken was set to true, an attacker could bypass the enforcement by sending any unknown agentKeyId — TryGetConnectToken returned false for the unknown ID, causing the code to skip token validation and allow the connection as a legacy agent. Added an else if (requireToken) branch inside the agentKeyId block: if agentKeyId is provided but not found in the AgentKeys.json registry in strict mode, the connection is now rejected with HubException("Unauthorized") and a warning is logged (ConnectAsDevice: unknown agentKeyId rejected). The attacker would still need the shared AES_MASTER_KEY_HEX to communicate, but the RequireConnectToken enforcement now correctly covers all cases.

Added: Share Link Expiry — Live Stream share links can now have a configurable lifetime. In the

stream toolbar, set Hours and Minutes or enable No time limit. The backend registers sessions in ShareSessionService; ShareExpiryService expires them and notifies viewers via ShareExpired. Admin UI shows a countdown while sharing. Server validates links on JoinShare and BroadcastSharedFrame (1 min – 7 days, or unlimited).

Added: Real-time Device Ping — DevicePingService pings connected agents every 10 s; agents respond with Pong; RTT is shown in the connected devices list (PingMs, color-coded). Cleartext ping IDs only (no sensitive payload).

Added: Telegram Notifications — Optional global setting in Agent settings: Telegram message when a device comes online. Settings persisted in telegram-settings.json (hot-reload); REST API api/telegram/settings + test endpoint; in-modal BotFather setup help.

Added: Telegram /Get_info Command — TelegramCommandService polls Bot API getUpdates; /Get_info requests Computer Info from an online agent (multi-device selection in chat). Responses routed via hub (SendComputerInfoRequestToAgentAsync, telegram- requestId prefix). Only the configured chat ID is accepted.

Fixed: Share Screen anonymous join — JoinShare added to hub anonymous allow-list so public viewers can connect without dashboard JWT.

Security: Replay Attack Protection (EncryptedPacket) — All incoming EncryptedPacket messages are now protected against replay attacks on both the BackEnd and RemoteAgent sides. A new ReplayCache (in-memory nonce-cache, BackEnd/Security/ReplayCache.cs and RemoteAgent/Security/ReplayCache.cs) registers the SHA-256 fingerprint of every received encrypted blob; an identical blob is rejected immediately. In addition, packet.Timestamp is validated against a configurable time window (default 60 seconds, tunable via ReplayProtection:MaxAgeSeconds in appsettings.json) — stale packets are dropped with a warning log. Validation runs in DeviceControlHub.ReceiveEncryptedFromAdmin, DeviceControlHub.ReceiveEncryptedFromAgent and SignalRService.HandleReceiveEncryptedPacketAsync. AES-GCM already guaranteed confidentiality and integrity; ReplayCache adds freshness so a captured valid ciphertext (e.g. a shutdown or kill-process command) cannot be replayed later.

OffCode - August 5, 2026, 4:57 am

Upcoming update!

Remote control android phones

- Gps tracking
- Live remote control
- File manager
- Open chat

OffCode - *September 25, 2026, 4:48 am*

Next week update is ready!

Android support!