



ISTITUTO NAZIONALE DI FISICA NUCLEARE

Sezione di Genova

INFN-2025-06-GE

14 Aprile 2025

INFN INTERNAL NOTES

Salvo Enrico

INFN, Sezione di Genova, I-16146 Genova, Italy

Abstract

Questa nota tecnica illustra la migrazione del sistema di accesso remoto utilizzato presso la Sezione INFN di Genova, passando da TightVNC a RustDesk. TightVNC, attualmente in uso, è un software ormai obsoleto, con limitazioni in termini di sicurezza, prestazioni e supporto per sistemi moderni. La crescente necessità di garantire un accesso remoto sicuro, efficiente e facilmente gestibile ha reso necessario valutare soluzioni alternative. RustDesk si è rivelato una scelta adeguata grazie alla sua natura open-source, alla possibilità di implementazione self-hosted e al supporto attivo della comunità. In questo documento vengono descritte le funzionalità principali di RustDesk, i motivi della sua adozione, la procedura di installazione personalizzata adottata presso la Sezione, e un confronto tecnico con TightVNC, evidenziando i vantaggi ottenuti in termini di sicurezza, usabilità e manutenzione.

PACS:07.05.Tp,89.20.Ff

*Published by
Laboratori Nazionali di Frascati*

1 Introduzione

Per fornire assistenza remota ai PC della Sezione, attualmente viene utilizzato TightVNC. Tuttavia, questo software presenta numerose criticità che ne rendono l'uso sempre meno adeguato in un contesto moderno:

- È un software obsoleto, sviluppato originariamente nel 2001.
- La password di accesso è limitata a soli 8 caratteri.
- Le credenziali sono memorizzate nel registro di sistema con una crittografia datata e facilmente vulnerabile.
- Il traffico tra client e server non è cifrato, esponendo i dati a potenziali intercettazioni.
- Utilizza un metodo di compressione video inefficiente e superato.
- Il supporto per sistemi Linux è stato interrotto nel 2009.
- Non supporta nativamente tecniche di NAT traversal, rendendone difficile l'utilizzo in scenari di smart working o reti complesse.

Queste limitazioni hanno reso necessaria la ricerca di una soluzione più moderna, sicura e flessibile. Anche alternative più recenti basate sul protocollo VNC, come TigerVNC, presentano problematiche significative in termini di prestazioni e licenze: per l'uso aziendale è spesso richiesto l'acquisto di una licenza, anche per software apparentemente open-source.

Per questi motivi si è deciso di abbandonare completamente l'ecosistema VNC, orientandosi verso una soluzione radicalmente diversa. RustDesk è un software open-source per l'accesso remoto, progettato per offrire un'alternativa moderna a strumenti come VNC, TeamViewer e AnyDesk. Una delle sue caratteristiche principali è la possibilità di autogestire l'infrastruttura server, garantendo così un maggiore controllo sui dati e una sicurezza superiore, in linea con le esigenze della Sezione.

2 Funzionalità Principali

RustDesk offre un insieme di funzionalità moderne pensate per garantire un accesso remoto sicuro, efficiente e facilmente gestibile. Tra le caratteristiche più rilevanti si segnalano:

- **Accesso remoto sicuro:** tutte le comunicazioni tra client e server sono protette tramite crittografia end-to-end, garantendo la riservatezza dei dati trasmessi.

- **Compatibilità multi-piattaforma:** il software è disponibile per i principali sistemi operativi, inclusi Windows, Linux e macOS, facilitando l'integrazione in ambienti eterogenei.
- **Connessione semplificata:** grazie alla disponibilità di un server di relay, è possibile stabilire connessioni anche in presenza di NAT o firewall, senza necessità di configurazioni complesse.
- **Trasferimento file:** consente lo scambio rapido e sicuro di file tra client e host, utile per attività di supporto e manutenzione.
- **Clipboard condivisa:** supporta il copia-incolla remoto, migliorando l'interazione tra sistemi e semplificando le operazioni di assistenza.
- **Interfaccia utente intuitiva:** l'interfaccia grafica è progettata per essere semplice e accessibile, rendendo l'utilizzo immediato anche per utenti non esperti.

3 Client RustDesk e gestione centralizzata

RustDesk non prevede una distinzione tra client e host: entrambi i nodi coinvolti nella connessione remota utilizzano lo stesso software. Questa simmetria semplifica l'installazione e la gestione, rendendo il sistema più flessibile.

Presso la Sezione INFN di Genova, l'installazione di RustDesk avviene tramite uno script PowerShell integrato nel sistema interno di installazione e aggiornamento automatico. Questo approccio consente di automatizzare la configurazione iniziale e di integrare la gestione centralizzata delle credenziali di accesso.

Durante l'installazione, ogni istanza di RustDesk genera un identificativo (ID) univoco. Questo ID viene trasmesso a un server centrale, che risponde assegnando una password di sistema. La password viene poi memorizzata nel database di configurazione centralizzata degli host, che gestisce anche altre informazioni come l'unità organizzativa (OU) di dominio e i parametri per l'avvio via PXE.

Un aspetto rilevante di questo sistema è la persistenza delle credenziali: la password assegnata viene mantenuta anche in caso di reinstallazione del client. Tuttavia, una nuova installazione genera un nuovo ID, permettendo così di rilevare eventuali reinstallazioni non autorizzate o anomale.

3.1 Gestione delle password

L'accesso remoto tramite RustDesk può avvenire utilizzando una delle due modalità di autenticazione previste:

- **Password amministrativa:** generata centralmente dal server e salvata nel database di configurazione. Questa password è gestita in modo centralizzato e consente un controllo uniforme sugli accessi da parte del personale autorizzato.

- **Password utente:** generata dinamicamente dalla GUI del client. Può essere reimpostata dall'utente in qualsiasi momento e viene rigenerata automaticamente a ogni disconnessione o cambio utente. Questa password è pensata per sessioni temporanee e non persiste oltre la durata della sessione attiva.

Le due modalità di autenticazione sono alternative: per stabilire una connessione è sufficiente disporre di una delle due password. È inoltre possibile configurare il client affinché accetti in modo permanente solo una delle due opzioni, in base alle policy di sicurezza adottate.

4 Infrastruttura server RustDesk

L'architettura di RustDesk prevede due componenti server principali, necessari per garantire il corretto funzionamento delle connessioni remote, in particolare in ambienti con NAT o firewall restrittivi. Presso la Sezione INFN di Genova, entrambi i server sono stati implementati in modalità self-hosted per garantire il pieno controllo sull'infrastruttura e sui dati.

4.1 hbbs – Signaling/Rendezvous Server

Il server hbbs svolge un ruolo centrale nel coordinamento delle comunicazioni tra i dispositivi client. Le sue principali funzioni includono:

- Gestione centralizzata degli ID RustDesk e associazione con gli indirizzi IP dei dispositivi.
- Gestione delle chiavi crittografiche utilizzate per la cifratura end-to-end delle comunicazioni.
- Supporto all'attraversamento NAT, facilitando la connessione diretta tra client anche in presenza di reti complesse.

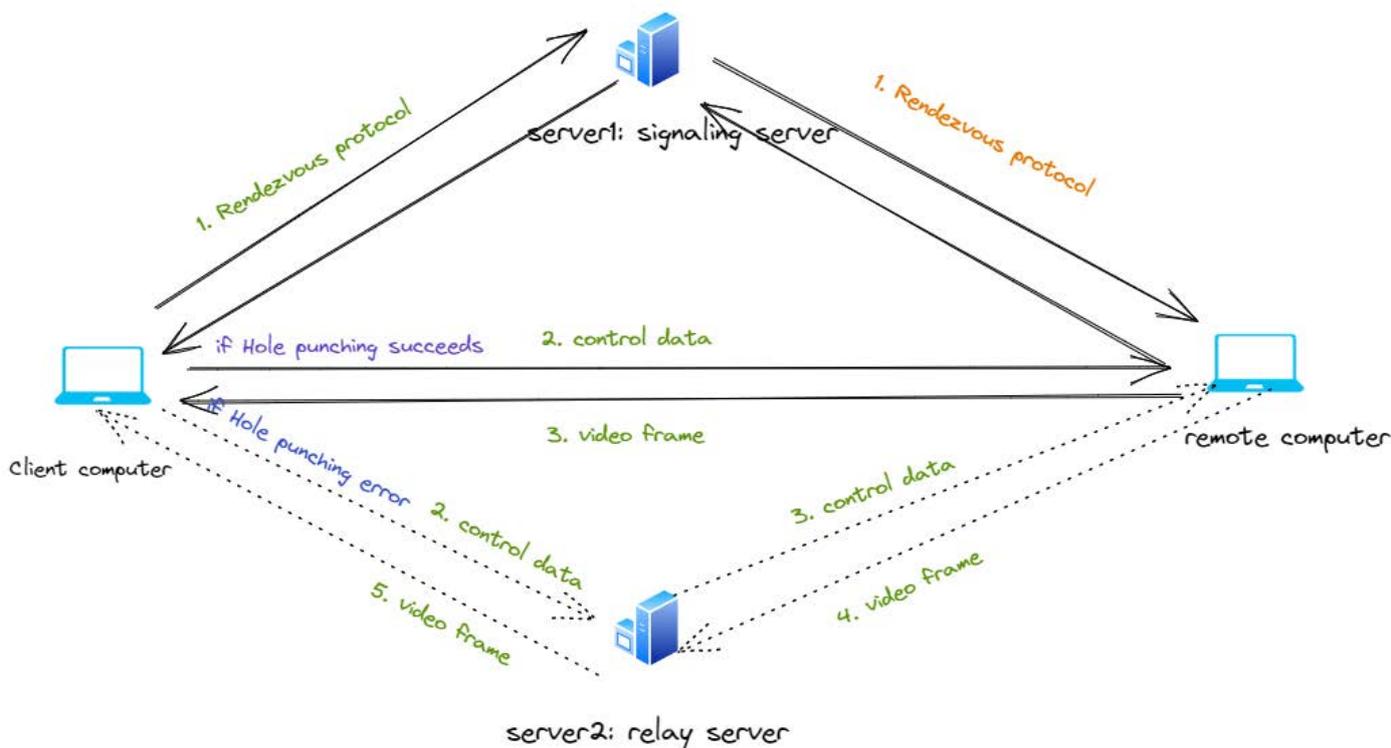
4.2 hbbr – Relay Server

Il server hbbr entra in funzione solo quando non è possibile stabilire una connessione diretta peer-to-peer tra i client, ad esempio a causa di firewall o NAT particolarmente restrittivi. In questi casi:

- Inoltra il traffico tra i dispositivi, fungendo da ponte per garantire la continuità della connessione.
- Introduce un leggero degrado delle prestazioni rispetto alla connessione diretta, ma assicura comunque l'operatività del sistema.

5 Architettura generale del sistema

L'architettura di RustDesk si basa su un modello client-server che prevede due componenti principali lato server: il signaling server (hbbs) e il relay server (hbbr). Il signaling server gestisce la fase iniziale di connessione tra i dispositivi, facilitando il rendezvous tra client e host, e coordinando lo scambio di dati di controllo e chiavi crittografiche. Quando la connessione diretta peer-to-peer è possibile (ad esempio tramite NAT traversal), il flusso video e i comandi di controllo vengono trasmessi direttamente tra i dispositivi. In caso contrario, entra in funzione il relay server, che inoltra i dati tra client e host, garantendo comunque la comunicazione, seppur con prestazioni leggermente inferiori. Questo schema consente una gestione flessibile e sicura delle connessioni remote, adattandosi dinamicamente alla configurazione di rete dei dispositivi coinvolti.



Note: Rendezvous protocol help to find each other

6 Licenze

RustDesk è disponibile in due versioni: una gratuita e una commerciale. La versione commerciale offre una serie di funzionalità avanzate pensate per ambienti enterprise, tra cui:

- Autenticazione centralizzata degli utenti tramite LDAP.
- Supporto per autenticazione a due fattori (2FA).
- Gestione centralizzata delle impostazioni di sicurezza sui client.

- API REST e interfaccia web per la configurazione e l'amministrazione del sistema.
- Logging avanzato delle attività dei client.
- Supporto per server relay multipli, utile in ambienti distribuiti.
- Supporto tecnico dedicato.

Attualmente, la Sezione INFN di Genova utilizza la versione gratuita. Alcune delle funzionalità mancanti sono state compensate attraverso l'integrazione con il sistema interno di installazione e gestione, che consente ad esempio la configurazione centralizzata delle impostazioni di sicurezza.

7 Confronto con TightVNC

Table 1: Confronto tra RustDesk e TightVNC

Caratteristica	RustDesk	TightVNC
Autenticazione	Due password (amministrativa e utente)	Una sola password (max 8 caratteri)
Sicurezza	Crittografia end-to-end (TLS 1.2)	Nessuna crittografia nativa
Qualità video	Streaming compresso con algoritmi moderni	Qualità video inferiore
Interfaccia	Intuitiva, moderna e user-friendly	Obsoleta e meno intuitiva

8 Conclusioni

RustDesk rappresenta una soluzione moderna, sicura e flessibile per l'accesso remoto, superando TightVNC sotto ogni aspetto: dalla sicurezza alla qualità del servizio, dalla gestione degli utenti alla compatibilità con reti complesse. L'adozione di un'infrastruttura self-hosted consente inoltre un controllo completo sull'ambiente operativo, in linea con le esigenze di sicurezza e autonomia dell'INFN.

References

- [1] RustDesk GitHub repository – documentazione ufficiale, codice sorgente e guida all'installazione: <https://github.com/rustdesk/rustdesk>
- [2] Documentazione ufficiale RustDesk – sezione dedicata all'installazione su Linux: <https://rustdesk.com/docs/en/client/linux/>
- [3] Wiki ufficiale RustDesk – panoramica sul funzionamento dell'architettura client-server: <https://github.com/rustdesk/rustdesk/wiki/How-does-RustDesk-work%3F>