

# 원격 **Vivado** 환경 설계: **Windows** 및 **macOS** 사용자를 위한 종합 가이드

## 요약

## 목표

본 보고서는 FPGA 엔지니어들이 Ubuntu 22.04 서버에 설치된 Xilinx Vivado 통합 설계 환경(IDE)에 Windows 11 또는 macOS 클라이언트에서 원격으로 접속할 수 있는 환경을 구축하기 위한 결정적인 가이드를 제공하는 것을 목표로 합니다.

## 핵심 문제

Vivado는 리소스를 많이 사용하는 그래픽 애플리케이션입니다. 원격 접속 환경을 구축하기 위해서는 성능(낮은 지연 시간, 높은 그래픽 충실도), 보안, 그리고 사용 편의성 사이의 균형을 맞춘 솔루션이 필요합니다.

## 방법론 개요

본 보고서는 세 가지 주요 원격 접속 프로토콜인 X11 포워딩, VNC(Virtual Network Computing), 그리고 RDP(Remote Desktop Protocol)를 심층적으로 분석합니다. 각 기술에 대한 상세하고 단계적인 구현 가이드를 제공하여 사용자가 자신의 요구사항에 가장 적합한 솔루션을 선택하고 구축할 수 있도록 지원합니다.

## 주요 권장 사항 요약

- 최상의 성능 및 기능 (특히 **WAN** 환경): xrdp를 통한 RDP가 가장 권장되는 솔루션입니다. RDP는 원시 픽셀 데이터가 아닌 드로잉 명령을 전송하는 효율적인 프로토콜을 사용하여 대역폭 사용량을 낮추고 반응성을 향상시킵니다.<sup>1</sup>
- 플랫폼 간 단순성 및 직접적인 화면 미러링 (주로 **LAN** 환경): SSH 터널을 통해 보안이 확보된 VNC는 가장 간단한 설정 방법을 제공하며 서버의 물리적 디스플레이를 직접 미러링합니다. 이는 기술 지원이나 공유 세션에 이상적입니다.<sup>1</sup>
- 가볍고 개별적인 애플리케이션 접속 (**LAN** 환경 전용): X11 포워딩은 단일 GUI 도구를 실행하는 데 적합하지만, 네트워크 지연 시간에 매우 민감하여 WAN을 통해 전체 Vivado IDE를 실행하는 데는 권장되지 않습니다.<sup>6</sup>

## 보고서 구조

본 보고서는 총 4부로 구성됩니다. 1부에서는 원격 기술을 비교 분석하고, 2부에서는 서버 측 구성 방법을, 3부에서는 클라이언트 측 구성 방법을 상세히 설명합니다. 마지막 4부에서는 성능 최적화, 보안 강화, 그리고 최종 권장 사항을 포함한 고급 주제를 다룹니다.

---

## 1부: 원격 그래픽 기술 비교 분석

이 파트에서는 각 기술의 근본적인 원리를 설명하여 사용자가 정보에 기반한 결정을 내릴 수 있도록 이론적 토대를 마련합니다.

### 1장: 원격 리눅스 **GUI**의 세 가지 핵심 기술

#### 1.1. X 윈도우 시스템과 **X11** 포워딩

### 핵심 원리

X11 포워딩의 기반이 되는 X 윈도우 시스템은 독특한 클라이언트-서버 모델을 사용합니다. 이 모델에서 "X 서버"는 사용자의 로컬 클라이언트 머신(Windows 또는 macOS)에서 실행되며, 그래픽 디스플레이와 입력 장치를 관리합니다. 반면, "X 클라이언트"는 원격 서버에서 실행되는 그래픽 애플리케이션(여기서는 **Vivado**)을 의미합니다.<sup>9</sup> 이러한 용어는 일반적인 클라이언트-서버 개념과 반대여서 혼란을 줄 수 있지만, 애플리케이션이 로컬 디스플레이 서버에 그래픽 렌더링을 "요청"하는 구조로 이해하면 명확해집니다. 이 모델의 핵심은 네트워크 투명성으로, X 클라이언트와 X 서버가 로컬 네트워크든 원격 네트워크든 동일한 방식으로 통신할 수 있도록 설계되었습니다.<sup>9</sup>

### 작동 방식

X11 포워딩은 SSH(Secure Shell) 연결을 통해 안전하게 구현됩니다. 사용자가 SSH 클라이언트에서 -X 또는 -Y 옵션을 사용하여 원격 서버에 접속하면, SSH는 암호화된 터널을 생성합니다.<sup>11</sup> 이 터널을 통해 원격 서버에서 실행되는 **Vivado**와 같은 X 클라이언트의 그래픽 명령이 로컬 머신의 X 서버로 안전하게 전달됩니다. 이 과정에서 SSH는 원격 서버의 DISPLAY 환경 변수를 자동으로 설정합니다. 이 변수는 일반적으로 localhost:10.0과 같은 값을 가지며, 이는 그래픽 출력을 로컬 머신의 특정 디스플레이로 다시 보내도록 지시하는 역할을 합니다.<sup>9</sup> 즉, 모든 그래픽 데이터는 암호화된 SSH 채널을 통과하므로 인터넷과 같은 비보안 네트워크에서도 안전하게 사용할 수 있습니다.

### 사용자 경험

X11 포워딩의 가장 큰 특징은 개별 애플리케이션 창을 로컬 데스크톱 환경에 직접 통합한다는 점입니다. **Vivado**를 실행하면, 전체 원격 데스크톱이 하나의 창에 표시되는 것이 아니라 **Vivado** 애플리케이션 창 자체가 마치 로컬에 설치된 프로그램처럼 나타납니다.<sup>6</sup> 사용자는 원격 애플리케이션 창을 로컬 애플리케이션 창과 나란히 배치하고, 크기를 조절하며, 최소화하는 등 자연스럽게 상호작용할 수 있습니다. 이는 여러 원격 시스템의 다양한 애플리케이션을 동시에 사용하는 시나리오에서 매우 강력한 장점이 됩니다.<sup>6</sup>

## 1.2. VNC (Virtual Network Computing)

### 핵심 원리

VNC는 X11과 근본적으로 다른 접근 방식을 취합니다. VNC는 프레임버퍼(framebuffer) 수준에서 작동하는 "원격 화면" 기술입니다.<sup>14</sup> VNC 서버는 원격 데스크톱 화면의 픽셀 데이터를 그대로 캡처하여 이 비트맵 이미지를 네트워크를 통해 VNC 클라이언트로 전송합니다.<sup>16</sup> 클라이언트는 수신된 픽셀 데이터를 화면에 렌더링하고, 사용자의 키보드 및 마우스 입력은 다시 서버로 전송되어 원격 컴퓨터를 제어합니다. 본질적으로 VNC는 원격 컴퓨터의 화면을 비디오로 스트리밍하는 것과 유사한 개념입니다.

### 작동 방식

VNC는 RFB(Remote Frame Buffer) 프로토콜을 기반으로 합니다.<sup>14</sup> VNC 서버는 원격 화면을 작은 사각형 영역으로 나누고, 이전 프레임과 비교하여 변경된 부분의 픽셀 데이터만 클라이언트로 전송합니다.<sup>15</sup> 이 방식은 마우스 포인터 이동이나 텍스트 입력과 같이 화면의 작은 부분만 변경될 때 효율적입니다. 하지만 창을 스크롤하거나 동영상 재생하는 등 화면

전체가 빠르게 변경되면 전송해야 할 데이터 양이 급격히 증가하여 대역폭을 많이 소모하게 됩니다.<sup>16</sup> 클라이언트와 서버는 연결 시 서로 지원하는 인코딩 방식을 협상하여 네트워크 상태에 가장 적합한 압축 기술을 사용합니다.

#### 사용자 경험

VNC를 사용하면 로컬 머신에 하나의 창이 열리고, 그 안에 원격 서버의 전체 데스크톱 환경이 표시됩니다.<sup>1</sup> 이는 X11 포워딩처럼 로컬 데스크톱과 통합되지는 않지만, 사용자에게 완전하고 독립적인 원격 작업 환경을 제공합니다. 사용자는 원격 데스크톱의 모든 요소(작업 표시줄, 아이콘, 애플리케이션 등)를 마치 그 컴퓨터 앞에 앉아 있는 것처럼 보고 제어할 수 있습니다. VNC는 서버의 물리적 디스플레이(일반적으로 :0)를 미러링하거나, 별도의 가상 디스플레이(:1, :2 등)를 생성하여 여러 사용자가 동시에 독립적인 세션을 가질 수 있도록 설정할 수도 있습니다.<sup>2</sup>

### 1.3. RDP (Remote Desktop Protocol)와 xrdp

#### 핵심 원리

마이크로소프트에서 개발한 RDP는 VNC보다 한 단계 더 높은 수준에서 작동하는 고급 프로토콜입니다. RDP는 원시 픽셀 데이터를 전송하는 대신, "여기에 창을 그려라", "이 폰트로 이 텍스트를 렌더링해라"와 같은 그래픽 드로잉 기본 요소(primitives)를 전송합니다.<sup>2</sup> 클라이언트는 이 명령어를 받아 자신의 그래픽 처리 능력을 사용해 화면을 렌더링합니다. 이 방식은 클라이언트 측의 리소스를 활용하고, 데이터 전송 전에 그래픽 객체를 캐싱하며, 네트워크를 통해 전송되는 데이터 양을 크게 줄여줍니다. 이는 RDP가 VNC나 X11에 비해 낮은 대역폭과 높은 지연 시간을 가진 네트워크 환경에서 훨씬 뛰어난 성능을 보이는 이유입니다.

#### 작동 방식

리눅스 환경에서는 xrdp라는 오픈소스 서버가 RDP 연결을 가능하게 합니다. xrdp는 RDP 클라이언트(예: Windows의 원격 데스크톱 연결)로부터의 연결을 수신한 후, 이를 리눅스 백엔드 세션 관리자에게 전달하는 브리지 역할을 합니다.<sup>2</sup> 이 백엔드는 일반적으로 Xorg 세션이거나, 경우에 따라서는 내부적으로 VNC 세션을 사용할 수도 있습니다. 즉, xrdp는 RDP의 효율적인 프로토콜과 리눅스의 표준 그래픽 시스템을 결합한 하이브리드 솔루션입니다.

#### 사용자 경험

RDP는 각 사용자에게 완전히 격리된 새로운 가상 세션을 제공합니다.<sup>1</sup> 이는 마치 Windows 원격 데스크톱을 사용하는 것과 유사한 부드럽고 풍부한 기능의 경험을 제공합니다. 양방향 클립보드(텍스트뿐만 아니라 이미지 및 파일 전송 지원), 오디오 리디렉션, 프린터 및 드라이브 공유와 같은 고급 기능들을 지원하여 원격 작업을 훨씬 더 생산적으로 만듭니다.<sup>2</sup> Vivado와 같은 복잡한 애플리케이션을 사용할 때 이러한 기능들은 작업 효율성을 크게 향상시킬 수 있습니다.

각 프로토콜이 가상화하는 대상의 차이는 성능과 기능의 모든 차이를 설명하는 근본적인 원인이 됩니다. VNC는 디스플레이 하드웨어, 즉 프레임버퍼를 가상화합니다. RDP는 그래픽 드라이버 또는 API를 가상화합니다. X11은 윈도우 시스템 자체를 가상화합니다. 이러한

근본적인 설계 차이는 왜 RDP가 느린 네트워크에서 더 효율적인지를 명확히 보여줍니다. RDP는 렌더링 작업을 클라이언트에 위임하고 더 추상적이고 압축 가능한 데이터를 전송하기 때문입니다. 또한, X11이 왜 그렇게 "수다스럽고(chatty)" 지연 시간에 민감한지도 설명해 줍니다. 모든 작은 상호작용이 애플리케이션과 로컬 X 서버 간의 왕복 통신을 요구하기 때문입니다.<sup>7</sup> VNC의 성능은 화면의 변경되는 양과 비트맵 압축 효율에 직접적으로 좌우됩니다. 이 개념적 모델은 사용자가 각 프로토콜이 특정 상황에서 어떻게 작동할지 예측하는 데 도움을 줍니다.

## 2장: Vivado 환경에 대한 프로토콜 적합성 평가

### 2.1. 성능 심층 분석: 지연 시간 vs. 대역폭

Vivado와 같이 복잡하고 상호작용이 많은 GUI 애플리케이션을 원격으로 사용할 때, 네트워크의 지연 시간(latency)과 대역폭(bandwidth)은 사용자 경험에 결정적인 영향을 미칩니다. 각 프로토콜은 이 두 가지 제약 조건에 다르게 반응합니다.

- **X11 포워딩:** 이 프로토콜은 지연 시간에 극도로 민감합니다. X11은 본질적으로 동기식(synchronous)이며, 많은 작업이 클라이언트와 서버 간의 왕복 통신을 요구합니다.<sup>7</sup> Vivado가 사용하는 Qt와 같은 현대적인 GUI 킷은 안티앨리어싱된 폰트, 복잡한 위젯, 그라데이션 배경 등 정교한 그래픽 요소를 렌더링하기 위해 수많은 X11 요청을 생성합니다.<sup>8</sup> 이로 인해 네트워크 지연 시간이 조금만 길어져도(예: WAN 환경) 애플리케이션의 반응이 고통스러울 정도로 느려집니다. 따라서 X11 포워딩은 지연 시간이 매우 낮은 로컬 영역 네트워크(LAN) 환경에서만 실용적입니다.
- **VNC:** VNC는 픽셀 기반 전송 방식으로 인해 대역폭 소모가 큼니다.<sup>1</sup> 소스 코드 파일을 스크롤하거나, 시뮬레이션 파형을 보거나, 레이아웃을 확대/축소하는 등 화면에 큰 변화가 발생할 때마다 상당한 양의 데이터를 전송해야 합니다. 고해상도 디스플레이를 사용할수록 이 문제는 더욱 심각해집니다. VNC의 성능은 지연 시간보다는 가용 대역폭에 더 크게 좌우됩니다. 대역폭이 충분한 LAN 환경에서는 좋은 성능을 보일 수 있지만, 대역폭이 제한적인 WAN 환경에서는 화면 업데이트가 지연되거나 끊기는 현상이 발생할 수 있습니다.
- **RDP (xrdp):** RDP는 지연 시간과 대역폭 제약 모두에 가장 잘 견디는 프로토콜입니다. 드로잉 명령 전송, 고급 압축 알고리즘, 클라이언트 측 비트맵 캐싱과 같은 프로토콜 수준의 최적화 덕분에 네트워크를 통해 전송되는 실제 데이터 양이 현저히 적습니다.<sup>1</sup> 이로 인해 인터넷을 통해 Vivado를 사용하는 경우에도 상대적으로 부드럽고 반응성이 뛰어난 사용자 경험을 제공하여, 원격 개발 환경을 위한 가장 우수한 선택지가 됩니다.

## 2.2. 보안 고려 사항

원격 접속 환경에서 보안은 타협할 수 없는 요소입니다. 각 프로토콜은 보안을 구현하는 방식에 있어 중요한 차이점을 보입니다.

- **X11 포워딩:** SSH를 통해 터널링하는 것이 표준적인 사용 방식이므로, 본질적으로 안전합니다.<sup>10</sup> 모든 X11 트래픽은 SSH의 강력한 암호화 채널을 통과하므로 데이터가 가로채거나 변조로부터 보호됩니다.
- **VNC:** 기본적인 VNC 프로토콜 자체는 암호화 기능이 없거나 매우 취약합니다.<sup>4</sup> VNC 서버를 인터넷에 직접 노출하는 것은 심각한 보안 위험을 초래합니다. 따라서 VNC를 안전하게 사용하기 위해서는 전체 VNC 세션을 SSH 터널을 통해 캡슐화하는 것이 절대적으로 필수적입니다.<sup>21</sup> 이 방식은 VNC 트래픽을 SSH의 암호화된 채널로 보내어 보안을 확보합니다.
- **RDP (xrdp):** xrdp는 TLS(Transport Layer Security) 암호화를 내장하고 있어 별도의 SSH 터널 없이도 강력한 보안을 기본적으로 제공합니다.<sup>4</sup> 이는 설정을 간소화하면서도 높은 수준의 보안을 유지할 수 있게 해주는 큰 장점입니다.

## 2.3. 기능 및 사용성 비교

- **클립보드:** RDP는 텍스트, 이미지, 심지어 파일까지 지원하는 가장 강력한 양방향 클립보드 기능을 제공합니다.<sup>2</sup> VNC는 대부분의 경우 일반 텍스트만 지원하며, X11 포워딩의 클립보드 통합은 클라이언트와 서버의 설정에 따라 일관성이 없을 수 있습니다.
- **다중 사용자 지원:** RDP는 기본적으로 각 사용자를 위한 격리된 동시 세션을 지원하도록 설계되었습니다.<sup>1</sup> 반면, VNC는 기본적으로 단일 세션을 공유하는 방식이지만, 수동으로 설정하면 사용자별로 별도의 가상 세션을 생성할 수 있습니다.
- **클라이언트 소프트웨어:** Windows에는 RDP 클라이언트가 기본적으로 내장되어 있으며, macOS용으로는 Microsoft가 제공하는 우수한 공식 클라이언트가 있습니다. VNC는 모든 플랫폼을 위한 다양한 클라이언트가 존재하여 선택의 폭이 넓습니다. 반면, X11 포워딩을 사용하려면 Windows와 macOS에 별도의 X 서버(예: VcXsrv, MobaXterm, XQuartz)를 설치해야 하므로 설정이 다소 복잡해집니다.<sup>2</sup>

### 표 1: Vivado 원격 접속 프로토콜 비교 매트릭스

다음 표는 사용자가 자신의 요구사항에 가장 적합한 기술을 한눈에 파악할 수 있도록 주요

특징을 요약한 것입니다.

기능	X11 포워딩	VNC (SSH 터널링)	RDP (xrdp)
통합 방식	개별 창 (로컬 데스크톱과 통합)	전체 데스크톱 (단일 창)	전체 데스크톱 (단일 창)
성능 (LAN)	양호	양호	우수
성능 (WAN)	나쁨 / 사용 불가	보통 / 튜닝 가능	최상
대역폭 사용량	높음 / Chatty	매우 높음	낮음 / 효율적
지연 시간 민감도	매우 높음	낮음	보통
보안 (기본)	SSH 통해 안전	자체는 불안전	TLS 통해 안전
서버 설정 난이도	쉬움	보통	보통
클라이언트 설정 난이도	복잡함	쉬움	쉬움
기능 집합 (클립보드 등)	기본	기본 / 다양함	풍부함
다중 사용자 지원	암묵적	수동 설정 필요	내장
<b>Vivado</b> 최적 사용 사례	LAN에서 단일 도구 실행	LAN 접속 / 화면 공유	주 원격 개발 환경 (특히 WAN)

## 2부: Ubuntu 22.04 서버 구성

이 파트에서는 리눅스 서버를 준비하기 위한 핵심적인 실습 지침을 제공합니다. 각 방법은 모듈식으로 구성되어 있어 사용자는 하나 또는 모든 방법을 구현할 수 있습니다.

## 3장: 서버 기초 준비 작업

원격 접속 환경을 구축하기 전에, 모든 방법에 공통적으로 필요한 서버의 기초를 다지는 작업이 선행되어야 합니다. 이 단계는 안정적이고 안전한 원격 환경의 기반이 됩니다.

### 3.1. 시스템 업데이트

가장 먼저, 서버의 패키지 목록을 최신 상태로 업데이트하고 설치된 모든 패키지를 최신 버전으로 업그레이드해야 합니다. 이는 시스템의 보안과 안정성을 보장하고, 이후에 설치될 소프트웨어와의 호환성 문제를 예방하는 중요한 단계입니다. 터미널에서 다음 명령을 실행합니다:

```
Bash
```

```
sudo apt update && sudo apt upgrade
```

27

### 3.2. OpenSSH 서버 설치

SSH는 모든 원격 관리 작업의 핵심입니다. X11 포워딩과 VNC 보안 터널링은 SSH에 직접적으로 의존하며, RDP 설정 시에도 원격 터미널 접속을 위해 필요합니다. 다음 명령으로 OpenSSH 서버를 설치합니다:

```
Bash
```

```
sudo apt install openssh-server
```



설치가 완료되면 **SSH** 서비스가 자동으로 시작되고 시스템 부팅 시 활성화됩니다. 다음 명령으로 서비스 상태를 확인하여 정상적으로 실행 중인지 확인할 수 있습니다:

Bash

```
sudo systemctl status ssh
```

출력 결과에서 **active (running)**이라는 문구가 보이면 정상입니다.

### 3.3. 경량 데스크톱 환경(XFCE) 설치

Vivado 자체는 그래픽 리소스를 많이 사용하지만, 원격으로 접속할 데스크톱 환경(DE)까지 무거울 필요는 없습니다. Ubuntu 22.04의 기본 데스크톱 환경인 GNOME은 기능이 풍부하지만 상대적으로 많은 시스템 리소스를 소모합니다. 원격 데스크톱 환경에서는 리소스를 최대한 Vivado에 할당하는 것이 성능에 유리합니다. 따라서 XFCE와 같은 경량 데스크톱 환경을 설치하는 것이 좋습니다. XFCE는 가볍고 빠르면서도 사용자 친화적인 인터페이스를 제공하여 원격 접속 성능을 향상시키는 데 큰 도움이 됩니다.<sup>30</sup>

다음 명령으로 XFCE와 관련 유틸리티를 설치합니다:

Bash

```
sudo apt install xfce4 xfce4-goodies
```

### 3.4. UFW 방화벽 구성

UFW(Uncomplicated Firewall)는 Ubuntu의 기본 방화벽 관리 도구입니다. 서버 보안을 위해 방화벽을 활성화하고 필요한 포트만 허용하는 것이 중요합니다.

먼저, 방화벽을 활성화하기 전에 SSH 접속이 끊기지 않도록 SSH 포트(기본값 22)를 허용해야 합니다. 이는 원격 서버에 대한 접근성을 잃지 않기 위한 매우 중요한 예방 조치입니다.

Bash

```
sudo ufw allow ssh
```

27

또는 포트 번호를 직접 지정할 수도 있습니다:

Bash

```
sudo ufw allow 22/tcp
```

이제 방화벽을 활성화합니다:

Bash

```
sudo ufw enable
```

29

시스템은 방화벽을 활성화하면 기존 SSH 연결이 끊어질 수 있다는 경고를 표시할 수 있습니다. 이미 SSH 포트를 허용했으므로 y를 입력하고 계속 진행합니다. 이로써 서버의 기초 준비가 완료되었으며, 이제 각 원격 접속 방식에 맞는 구체적인 설정을 진행할 수 있습니다.

## 4장: 방법 A - X11 포워딩 구성

X11 포워딩은 SSH 서버의 간단한 설정 변경만으로 활성화할 수 있어 가장 빠르게 시도해볼 수 있는 방법입니다.

### 4.1. SSH 데몬 구성 편집

SSH 서버의 주 설정 파일인 `/etc/ssh/sshd_config`를 편집하여 X11 포워딩을 활성화해야 합니다. `sudo` 권한으로 텍스트 편집기(예: `nano`)를 사용하여 파일을 엽니다:

```
Bash
```

```
sudo nano /etc/ssh/sshd_config
```

36

파일 내에서 `X11Forwarding` 지시어를 찾습니다. 주석 처리되어 있거나 `no`로 설정되어 있다면, 주석(`#`)을 제거하고 값을 `yes`로 변경합니다:

```
X11Forwarding yes
```

10

보안 및 호환성을 위해 다음 두 줄도 확인하고 필요시 추가하거나 수정하는 것이 좋습니다:

```
X11DisplayOffset 10  
X11UseLocalhost yes
```

X11UseLocalhost를 **yes**로 설정하면 X11 포워딩 연결이 서버의 루프백 주소에 바인딩되어 다른 원격 클라이언트가 포워딩된 디스플레이에 접근하는 것을 방지하여 보안을 강화합니다.

## 4.2. SSH 서비스 재시작

설정 파일 변경 사항을 적용하려면 SSH 서비스를 재시작해야 합니다:

```
Bash
```

```
sudo systemctl restart ssh
```

## 4.3. 필수 X11 패키지 설치

X11 포워딩이 정상적으로 작동하려면 서버에 **xauth** 프로그램이 설치되어 있어야 합니다. 이 프로그램은 X11 세션에 대한 인증 쿠키를 관리하는 역할을 합니다. 또한, 연결을 테스트하기 위한 간단한 X11 애플리케이션을 설치하는 것이 유용합니다. **x11-apps** 패키지에는 **xeyes**와 **xclock** 같은 테스트용 프로그램이 포함되어 있습니다.<sup>10</sup>

다음 명령으로 필요한 패키지를 설치합니다:

```
Bash
```

```
sudo apt install xauth x11-apps
```

이제 서버는 X11 포워딩 연결을 수락할 준비가 되었습니다.

## 5장: 방법 B - 보안 VNC 서버(TigerVNC) 배포

VNC는 SSH 터널링과 결합될 때 강력하고 안전한 원격 데스크톱 솔루션이 됩니다. 여기서는 널리 사용되는 TigerVNC 서버를 기준으로 설명합니다.

### 5.1. TigerVNC 서버 설치

Ubuntu의 공식 저장소에서 TigerVNC 서버 패키지를 설치합니다:

```
Bash
```

```
sudo apt install tigervnc-standalone-server tigervnc-common
```

30

### 5.2. 초기 VNC 사용자 설정

VNC 서버를 처음 실행하기 전에, **Vivado**를 실행할 사용자로 로그인하여 초기 설정을 진행해야 합니다. 터미널에서 **vncserver** 명령을 실행합니다:

```
Bash
```

```
vncserver
```

30

이 명령을 처음 실행하면, VNC 세션에 접속할 때 사용할 비밀번호를 설정하라는 메시지가

나타납니다. 이 비밀번호는 6~8자 길이어야 하며, VNC 프로토콜 자체의 인증에 사용됩니다. 이후 읽기 전용 비밀번호를 설정할지 묻는데, 특별한 경우가 아니면 **n**을 선택하면 됩니다.

이 과정은 사용자의 홈 디렉터리 아래에 `~/.vnc` 디렉터리를 생성하고, 초기 설정 파일(`passwd`, `xstartup`)을 만듭니다. 또한, 테스트용 VNC 세션을 디스플레이 번호 `:1`로 시작합니다.

VNC 비밀번호는 종종 약한 암호화 방식으로 저장되므로, 이 인증 방식에만 의존하는 것은 매우 위험합니다. 이것이 바로 모든 VNC 연결을 강력한 SSH 인증을 사용하는 SSH 터널을 통해 강제하는 이유입니다.

초기 설정이 완료되었으므로, 방금 시작된 테스트 세션을 종료합니다:

```
Bash
```

```
vncserver -kill :1
```

40

### 5.3. XFCE용 `xstartup` 구성

`~/.vnc/xstartup` 파일은 VNC 세션이 시작될 때 실행할 명령을 정의하는 셸 스크립트입니다. 이 파일을 편집하여 우리가 설치한 XFCE 데스크톱 환경이 시작되도록 설정해야 합니다.<sup>30</sup>

텍스트 편집기로 `~/.vnc/xstartup` 파일을 엽니다:

```
Bash
```

```
nano ~/.vnc/xstartup
```

기존 내용을 모두 지우고 다음 내용으로 교체합니다. 이 스크립트는 기존 세션 관리자 설정을 초기화하고 XFCE 세션을 시작합니다<sup>30</sup>:

Bash

```
#!/bin/sh
unset SESSION_MANAGER
unset DBUS_SESSION_BUS_ADDRESS
exec startxfce4
```

파일을 저장한 후, 스크립트가 실행될 수 있도록 실행 권한을 부여합니다:

Bash

```
chmod +x ~/.vnc/xstartup
```

30

## 5.4. VNC 서버가 로컬호스트에서만 수신하도록 구성

보안 필수 조치: 모든 VNC 연결이 안전한 SSH 터널을 통과하도록 강제하려면, VNC 서버가 서버 자체, 즉 로컬호스트(`localhost`, `127.0.0.1`)로부터의 연결만 허용하도록 설정해야 합니다. 이렇게 하면 외부에서 VNC 포트로 직접 접속하는 것이 불가능해집니다.

VNC 서버를 시작할 때 `-localhost` 옵션을 추가하면 됩니다.<sup>22</sup> 예를 들어, 디스플레이

:1로 VNC 세션을 시작하려면 다음 명령을 사용합니다:

Bash

```
vncserver -localhost :1
```

## 5.5. 방화벽 구성

VNC 서버를 로컬호스트에서만 수신하도록 설정했기 때문에, UFW 방화벽에서 VNC 포트(5901 등)를 열 필요가 없습니다. 모든 통신은 이미 허용된 SSH 포트 22를 통해 이루어집니다. 이는 외부 공격에 대한 노출을 최소화하는 매우 중요한 보안 설정입니다.

## 6장: 방법 C - RDP 서버(xrdp) 배포

xrdp는 리눅스 서버에 RDP 접속을 가능하게 하여 Windows 사용자에게 친숙하고 성능이 뛰어난 원격 데스크톱 경험을 제공합니다.

### 6.1. xrdp 설치

다음 명령으로 xrdp를 설치합니다. -y 플래그는 설치 확인 프롬프트를 자동으로 수락합니다:

```
Bash
```

```
sudo apt install xrdp -y
```

18

설치가 완료되면 xrdp 서비스가 자동으로 시작됩니다. 다음 명령으로 상태를 확인할 수 있습니다:

```
Bash
```

```
sudo systemctl status xrdp
```

44



## 6.2. xrdp 사용자를 ssl-cert 그룹에 추가

xrdp는 TLS 암호화를 위해 SSL 인증서 개인 키를 읽어야 합니다. 기본적으로 이 키 파일은 root 사용자만 읽을 수 있어 xrdp가 접근할 때 권한 오류가 발생할 수 있습니다. 이 문제를 해결하기 위해 xrdp 사용자를 ssl-cert 그룹에 추가하여 필요한 읽기 권한을 부여합니다. 이는 xrdp 연결 실패의 흔한 원인을 해결하는 중요한 단계입니다.<sup>45</sup>

Bash

```
sudo adduser xrdp ssl-cert
```

## 6.3. 세션 관리자 구성

RDP로 로그인할 때 어떤 데스크톱 환경을 시작할지 xrdp에 알려주어야 합니다. 이를 위해 접속하려는 사용자의 홈 디렉터리에 .xsession 파일을 생성하고, 그 안에 XFCE 세션을 시작하는 명령을 기입합니다:

Bash

```
echo "xfce4-session" > ~/.xsession
```

32

이 명령은 현재 사용자의 홈 디렉터리에 .xsession 파일을 생성하고 xfce4-session이라는 내용을 씁니다. 다른 사용자에 대해서도 이 설정을 적용해야 합니다.

## 6.4. "검은 화면" 문제와 GNOME 호환성 문제 해결

Ubuntu 22.04 데스크톱 버전을 설치한 경우, 기본 데스크톱 환경인 GNOME은 Wayland라는 디스플레이 서버를 사용합니다. 하지만 xrdp는 현재 Wayland와 호환되지 않으며, 전통적인 Xorg 디스플레이 서버가 필요합니다.<sup>32</sup> 이 불일치는

xrdp 사용자가 겪는 가장 흔한 문제 중 하나로, 로그인에는 성공하지만 이후 검은 화면만 보이거나 연결이 즉시 끊기는 증상으로 나타납니다.<sup>32</sup>

이 가이드에서는 Xorg를 기본으로 사용하는 XFCE를 설치하여 이 문제를 근본적으로 회피했습니다. 따라서 XFCE를 사용하는 경우에는 추가 조치가 필요 없습니다. 하지만 이 문제의 원인을 이해하는 것은 다른 환경에서 발생할 수 있는 문제를 해결하는 데 중요합니다. 만약 GNOME 데스크톱을 꼭 사용해야 한다면, `/etc/gdm3/custom.conf` 파일을 편집하여 `WaylandEnable=false` 줄의 주석을 해제함으로써 Wayland를 비활성화하고 GDM(GNOME Display Manager)을 재시작해야 합니다.<sup>32</sup> 또는,

`/etc/xrdp/startwm.sh` 파일에 특정 환경 변수를 추가하여 GNOME 세션이 Xorg 기반으로 실행되도록 유도하는 방법도 있습니다.<sup>47</sup>

## 6.5. 방화벽 구성

마지막으로 UFW 방화벽에서 RDP가 사용하는 기본 포트인 3389를 허용해야 합니다.

Bash

```
sudo ufw allow 3389/tcp
```

49

이제 서버는 RDP 클라이언트로부터의 연결을 수락할 준비가 되었습니다.

---

## 3부: 클라이언트 환경 구성

이 파트에서는 사용자의 로컬 머신에서 서버에 접속하기 위한 설정 방법을 Windows 11과

macOS에 대해 각각 설명합니다.

## 7장: Windows 11에서 연결

### 7.1. 방법 A: MobaXterm을 이용한 X11 포워딩

이론적 근거

Windows에서 X11 포워딩을 설정하는 가장 간편한 방법은 MobaXterm을 사용하는 것입니다. MobaXterm은 SSH 클라이언트, 내장 X 서버, SFTP 브라우저 등 원격 작업에 필요한 모든 도구를 하나의 패키지로 제공하여 복잡한 설정을 크게 단순화합니다.<sup>51</sup>

설정 절차

1. MobaXterm 공식 웹사이트에서 Home Edition을 다운로드하여 설치합니다.<sup>54</sup>
2. MobaXterm을 실행하고, 상단 메뉴에서 'Session'을 클릭합니다.
3. 세션 유형으로 'SSH'를 선택합니다.
4. 'Remote host' 필드에 Ubuntu 서버의 IP 주소를 입력하고, 'Specify username'을 체크한 후 사용자 이름을 입력합니다.
5. 'Advanced SSH settings' 탭으로 이동하여 'X11-Forwarding'이 체크되어 있는지 확인합니다. MobaXterm에서는 이 옵션이 기본적으로 활성화되어 있습니다.<sup>52</sup>
6. 'OK'를 클릭하여 세션을 저장하고 연결을 시작합니다.
7. 비밀번호를 입력하여 로그인한 후, 터미널에서 `xeyes`와 같은 그래픽 애플리케이션을 실행하여 테스트합니다. 잠시 후 눈 모양의 창이 Windows 데스크톱에 나타나면 성공입니다. 그 다음 `vivado`를 실행하면 됩니다.

### 7.2. 방법 B: VcXsrv와 PuTTY를 이용한 X11 포워딩

설정 절차

이 방법은 각 구성 요소를 개별적으로 설치하고 설정하는 수동 방식입니다.

1. **VcXsrv 설치:** Windows용 X 서버인 VcXsrv를 SourceForge에서 다운로드하여 설치합니다.<sup>51</sup> 설치 후 'XLaunch'를 실행하여 기본 설정으로 X 서버를 시작합니다. 작업 표시줄 트레이에 VcXsrv 아이콘이 나타나면 정상적으로 실행 중인 것입니다.
2. **PuTTY 설치:** SSH 클라이언트인 PuTTY를 공식 웹사이트에서 다운로드하여 설치합니다.<sup>58</sup>
3. **PuTTY 설정:**
  - PuTTY를 실행하고 'Session' 카테고리에서 서버의 IP 주소를 'Host Name'에

입력합니다.

- 왼쪽 카테고리 트리에서 **Connection > SSH > X11**로 이동합니다.
  - 'Enable X11 forwarding' 체크박스를 선택합니다.<sup>56</sup>
  - 다시 'Session' 카테고리로 돌아와서, 'Saved Sessions' 아래에 이름을 입력하고 'Save' 버튼을 눌러 설정을 저장합니다.
4. 연결: 'Open' 버튼을 클릭하여 서버에 연결하고, 로그인 후 그래픽 애플리케이션을 실행하여 테스트합니다.

### 7.3. 방법 C: TigerVNC 뷰어와 PuTTY를 이용한 VNC over SSH

설정 절차

1. **TigerVNC** 뷰어 설치: TigerVNC 공식 웹사이트 또는 GitHub 릴리즈 페이지에서 Windows용 뷰어(vncviewer64-x.x.x.exe)를 다운로드하여 설치합니다.<sup>61</sup>
2. **PuTTY SSH** 터널 설정:
  - PuTTY를 실행하고 서버 접속 정보를 입력합니다.
  - 왼쪽 카테고리 트리에서 **Connection > SSH > Tunnels**로 이동합니다.
  - 'Source port'에 5901을 입력합니다.
  - 'Destination'에 localhost:5901을 입력합니다.
  - 'Add' 버튼을 클릭하여 터널 설정을 추가합니다.<sup>23</sup>
3. 연결: PuTTY 세션을 저장하고 서버에 연결합니다. 로그인하면 SSH 터널이 활성화됩니다. 이 PuTTY 창을 열어두어야 터널이 유지됩니다.
4. **VNC** 접속: TigerVNC 뷰어를 실행하고 'VNC Server' 필드에 localhost:5901 또는 localhost:1을 입력한 후 'Connect'를 클릭합니다.<sup>61</sup> VNC 서버 설정 시 입력했던 비밀번호를 입력하면 원격 데스크톱 화면이 나타납니다.

### 7.4. 방법 D: 내장 클라이언트를 이용한 RDP

설정 절차

1. Windows 시작 메뉴에서 '원격 데스크톱 연결'을 검색하거나 mstsc.exe를 실행합니다.<sup>49</sup>
2. '컴퓨터' 필드에 Ubuntu 서버의 IP 주소를 입력합니다.
3. '연결'을 클릭합니다.
4. xrdp 로그인 창이 나타나면, Ubuntu 사용자의 이름과 비밀번호를 입력하고 'OK'를 클릭합니다.<sup>66</sup>
5. 연결이 성공하면 원격 XFCE 데스크톱 환경이 나타납니다.

## 8장: macOS에서 연결

### 8.1. 방법 A: XQuartz를 이용한 X11 포워딩

#### 설정 절차

1. **XQuartz 설치:** macOS용 공식 X 서버인 XQuartz를 공식 웹사이트에서 다운로드하여 설치합니다.<sup>67</sup> 설치 후에는 시스템을 재시동하거나 로그아웃 후 다시 로그인해야 할 수 있습니다.
2. 연결: 내장 '터미널' 앱을 엽니다.
3. 다음 명령을 사용하여 SSH를 통해 서버에 연결합니다. -Y 옵션은 신뢰할 수 있는 X11 포워딩을 활성화하며, 일부 애플리케이션에서 -X보다 호환성이 좋습니다.<sup>67</sup>

Bash

```
ssh -Y 사용자이름@서버_IP_주소
```

4. 테스트: 로그인 후, 터미널에서 **xeyes** 또는 **vivado**를 실행합니다. XQuartz가 자동으로 실행되고 애플리케이션 창이 macOS 데스크톱에 나타납니다.

### 8.2. 방법 B: 내장 화면 공유 클라이언트를 이용한 VNC over SSH

#### 설정 절차

1. **SSH 터널 설정:** '터미널' 앱을 열고 다음 명령을 실행하여 SSH 터널을 설정합니다. 이 터널은 로컬 포트 5901로 들어오는 모든 트래픽을 SSH 연결을 통해 서버의 5901 포트로 안전하게 전달합니다.

Bash

```
ssh -L 5901:localhost:5901 사용자이름@서버_IP_주소
```

23

로그인 후 이 터미널 창을 열어두어야 터널이 유지됩니다.

2. 화면 공유 앱 실행: **Finder**를 열고 메뉴 막대에서 '이동' > '서버에 연결...'을 선택하거나, 단축키 **Cmd+K**를 누릅니다.<sup>72</sup>
3. 접속: 서버 주소 필드에 **vnc://localhost:5901**을 입력하고 '연결'을 클릭합니다.<sup>72</sup>

4. 비밀번호를 묻는 창이 나타나면, VNC 서버 설정 시 지정했던 비밀번호를 입력합니다.

### 8.3. 방법 C: Microsoft Remote Desktop for Mac을 이용한 RDP

#### 설정 절차

1. 클라이언트 설치: Mac App Store에서 'Microsoft Remote Desktop'을 검색하여 설치합니다.<sup>50</sup>
  2. PC 연결 추가: 앱을 실행하고 '+' 버튼을 클릭한 후 'Add PC'를 선택합니다.
  3. 'PC name' 필드에 Ubuntu 서버의 IP 주소를 입력합니다. 다른 설정은 기본값으로 두어도 무방합니다.
  4. 'Add'를 클릭하여 PC를 추가합니다.
  5. 생성된 PC 아이콘을 더블 클릭하여 연결을 시작합니다.
  6. Ubuntu 사용자의 이름과 비밀번호를 입력하라는 메시지가 나타나면 정보를 입력하고 'Continue'를 클릭합니다.<sup>76</sup>
- 

## 4부: 고급 주제 및 권장 사항

이 마지막 파트는 단순한 가이드를 넘어, 실제 환경에서 마주할 수 있는 중요한 과제들을 해결하는 전문가 수준의 참고 자료를 제공합니다.

## 9장: 성능 최적화 및 튜닝

### 9.1. 일반 원칙

원격 데스크톱 성능을 튜닝할 때는 시각적 품질, 응답성, 그리고 네트워크/**CPU** 부하라는 세 가지 요소 사이의 상충 관계를 이해하는 것이 중요합니다. 하나의 요소를 최적화하면 다른 요소가 저하될 수 있으므로, 사용자의 네트워크 환경과 작업 요구사항에 맞는 최적의 균형점을 찾아야 합니다.

## 9.2. Vivado 자체 최적화

Vivado는 CPU와 RAM을 많이 사용하는 애플리케이션이며, 특히 단일 코어 성능이 전체 작업 속도에 큰 영향을 미칩니다.<sup>78</sup> 따라서 원격 서버 자체의 사양이 높아야 합니다.

또한, 모든 작업을 그래픽 인터페이스를 통해 수행할 필요는 없습니다. 합성과 구현 같은 장시간 실행되는 작업은 GUI 없이 실행하는 것이 훨씬 효율적입니다. Tcl(Tool Command Language) 스크립트를 사용하여 Vivado를 배치 모드로 실행하면 원격 데스크톱의 오버헤드를 완전히 제거할 수 있습니다.

- Tcl 셸 모드: `vivado -mode tcl`
- 배치 모드: `vivado -mode batch -source <your_script.tcl>`

79

이러한 비-GUI 워크플로우는 리소스를 절약하고 자동화된 빌드 파이프라인을 구축하는 데 필수적입니다.

## 9.3. X11 포워딩 튜닝

X11 포워딩의 성능은 현대적인 애플리케이션의 비효율적인 렌더링 방식 때문에 근본적인 한계를 가집니다. Vivado가 사용하는 Qt와 같은 최신 GUI 킷은 안티앨리어싱된 폰트나 복잡한 위젯을 렌더링할 때, X11의 드로잉 명령 대신 거대한 비트맵 이미지를 전송하는 경우가 많습니다.<sup>20</sup> 이는 X11 프로토콜의 본래 장점을 무력화시키고 대역폭 사용량을 급증시키는 주요 원인입니다. 이로 인해 빠른 LAN 환경에서도 일부 현대적인 애플리케이션이 X11을 통해 느리게 느껴질 수 있습니다.

그럼에도 불구하고, 몇 가지 튜닝을 시도해 볼 수 있습니다:

- **SSH 압축 활성화:** SSH 연결 시 `-C` 플래그를 추가하면 전송되는 데이터를 압축합니다 (`ssh -C -Y...`).<sup>82</sup> 이는 대역폭이 낮은 네트워크에서는 도움이 될 수 있지만, 빠른 네트워크에서는 압축/해제에 따른 CPU 오버헤드로 인해 오히려 지연 시간이 증가할 수 있습니다.
- **빠른 암호화 방식 사용:** `aes128-gcm@openssh.com`과 같은 최신 암호화 방식을 지정하면 암호화 오버헤드를 줄일 수 있습니다 (`ssh -c aes128-gcm@openssh.com...`).<sup>83</sup>
- **데스크톱 효과 비활성화:** 서버 측 XFCE 데스크톱 환경에서 창 그림자나 애니메이션 같은 시각 효과(컴포지팅)를 비활성화하면 전송해야 할 그래픽 데이터의 복잡성을 줄여 성능 향상에 도움이 될 수 있습니다.<sup>20</sup>

## 9.4. VNC 튜닝

VNC는 다양한 옵션을 통해 성능을 조절할 수 있습니다.

- 인코딩 방식: **Tight**나 **ZRLE**와 같은 효율적인 인코딩 방식을 사용하면 데이터 전송량을 줄일 수 있습니다.<sup>84</sup>
- 색상 깊이: 색상 깊이를 24비트(True Color)에서 16비트(High Color)나 8비트(256 색상)로 낮추면 대역폭 요구량을 극적으로 줄일 수 있습니다.<sup>86</sup> Vivado 작업에 항상 최고 화질이 필요한 것은 아니므로, 이는 가장 효과적인 최적화 방법 중 하나입니다.
- 압축 수준: 압축 수준과 JPEG 품질을 조절할 수 있습니다. 압축률을 높이면 대역폭은 절약되지만 서버의 CPU 사용량이 증가하여 지연 시간이 발생할 수 있습니다. 반대로 압축률을 낮추면 CPU 부담은 줄지만 더 많은 대역폭이 필요합니다.<sup>85</sup>

## 9.5. RDP (xrdp) 튜닝

RDP는 이미 효율적이지만, 몇 가지 추가적인 튜닝으로 성능을 더욱 향상시킬 수 있습니다.

- 클라이언트 측 설정: Windows의 '원격 데스크톱 연결' 클라이언트나 macOS용 클라이언트에서 연결 설정을 변경할 수 있습니다. '디스플레이' 탭에서 해상도와 색상 깊이(16비트로 설정 권장)를 낮추고, '성능' 탭에서 '연결 속도'를 낮은 대역폭으로 설정하면 폰트 스무딩이나 데스크톱 배경과 같은 불필요한 시각 효과가 비활성화되어 성능이 향상됩니다.<sup>32</sup>
- 서버 측 데스크톱 환경: 가장 효과적인 서버 측 최적화는 XFCE나 LXDE와 같은 경량 데스크톱 환경을 사용하는 것입니다. 이는 xrdp 세션의 리소스 사용량을 최소화하여 Vivado에 더 많은 자원을 할당할 수 있게 합니다.<sup>32</sup>

# 10장: 보안 강화 및 모범 사례

원격 서버를 운영할 때는 보안을 최우선으로 고려해야 합니다. 다음은 원격 Vivado 환경을 안전하게 보호하기 위한 필수적인 조치들입니다.

## 10.1. SSH 키 기반 인증 필수 적용



비밀번호 인증은 무차별 대입 공격(brute-force attack)에 취약합니다. SSH 키 기반 인증은 훨씬 더 안전한 대안을 제공합니다.

1. 키 쌍 생성: 로컬 머신에서 `ssh-keygen` 명령을 사용하여 공개 키와 개인 키 쌍을 생성합니다.<sup>87</sup>
2. 공개 키 배포: `ssh-copy-id` 유틸리티를 사용하여 로컬 머신의 공개 키를 원격 서버의 `~/.ssh/authorized_keys` 파일에 안전하게 복사합니다.<sup>87</sup>
3. 비밀번호 인증 비활성화: SSH 키 기반 로그인이 정상적으로 작동하는 것을 확인한 후, 서버의 `/etc/ssh/sshd_config` 파일을 편집하여 `PasswordAuthentication no`로 설정합니다. 이 설정은 비밀번호를 이용한 로그인을 완전히 차단하여 서버 보안을 크게 강화합니다.<sup>87</sup>

## 10.2. 최소 권한 원칙: 루트 로그인 비활성화

SSH를 통해 `root` 사용자로 직접 로그인하는 것은 매우 위험한 관행입니다. 공격자가 `root` 계정 접근에 성공하면 시스템 전체를 장악할 수 있습니다. `/etc/ssh/sshd_config` 파일에서 `PermitRootLogin no`로 설정하여 `root` 로그인을 비활성화하고, 항상 일반 사용자 계정으로 로그인한 후 필요할 때 `sudo`를 사용하여 관리자 권한을 얻는 것이 안전합니다.<sup>87</sup>

## 10.3. UFW를 이용한 세분화된 방화벽 제어

단순히 포트를 여는 것 이상으로, 특정 IP 주소나 서브넷에서만 접속을 허용하도록 방화벽 규칙을 세분화해야 합니다. 신뢰할 수 있는 네트워크(예: 사무실이나 집의 고정 IP)에서만 접속을 허용하면 공격에 노출될 위험을 크게 줄일 수 있습니다.

- **SSH 접속 제한:**

Bash

```
sudo ufw allow from <your_trusted_ip> to any port 22
```

34

- **RDP 접속 제한:**

Bash

```
sudo ufw allow from <your_trusted_ip> to any port 3389
```

## 10.4. VNC 및 RDP 보안

- **VNC:** 앞서 강조했듯이, VNC는 반드시 SSH 터널을 통해 사용해야 하며, VNC 서버 자체는 localhost에서만 연결을 수신하도록 설정해야 합니다.<sup>22</sup> 이렇게 하면 암호화되지 않은 VNC 트래픽이 외부 네트워크에 노출되는 것을 원천적으로 차단할 수 있습니다.
- **RDP (xrdp):** xrdp의 보안을 강화하려면 /etc/xrdp/xrdp.ini 파일에서 TLS 관련 설정을 확인해야 합니다. security\_layer를 tls로 설정하고, ssl\_protocols에 TLSv1.2와 TLSv1.3 같은 최신 보안 프로토콜만 명시하여 구형의 취약한 프로토콜 사용을 금지하는 것이 좋습니다.<sup>25</sup>

## 11장: 최종 권장 사항 및 결론

### 11.1. 의사 결정 흐름도 / 요약

사용자의 상황에 가장 적합한 솔루션을 선택하는 데 도움이 되는 간단한 의사 결정 가이드입니다.

- 주로 집이나 외부(**WAN**)에서 작업해야 합니까?
  - 예: **\*\*RDP (xrdp)\*\***를 사용하십시오. WAN 환경에서 가장 뛰어난 성능과 안정성을 제공합니다.
- 빠르고 안전한 내부 네트워크(**LAN**)에서 작업합니까?
  - 최상의 경험을 원한다면: **RDP**가 여전히 가장 좋은 선택입니다.
  - 간단한 설정이나 동료와의 화면 공유가 필요하다면: **VNC**가 좋은 대안입니다.
  - 단일 도구만 가볍게 실행하고 싶다면: **X11** 포워딩을 시도해볼 수 있습니다.
- 자신의 화면을 다른 사람에게 그대로 보여주어야 합니까?
  - 예: 물리적 디스플레이를 미러링하는 **VNC**를 사용하십시오.
- 원격 애플리케이션 창을 로컬 데스크톱에 완벽하게 통합하고 싶습니까?
  - 예: LAN 환경에서 **X11** 포워딩을 시도해 보십시오.

### 11.2. Vivado 관련 워크플로우 권장 사항

원격 환경의 장점을 극대화하고 단점을 최소화하기 위한 Vivado 작업 흐름입니다.

- 장시간 실행 작업: 합성, 구현, 비트스트림 생성과 같이 시간이 오래 걸리는 작업은 그래픽 인터페이스 없이 일반 SSH 세션에서 Vivado를 Tcl 스크립트나 배치 모드로 실행하는 것이

가장 효율적입니다.

- 대화형 작업: 설계 입력, 시뮬레이션 파형 분석, 제약 조건 편집 등 상호작용이 필요한 작업에 원격 그래픽 세션을 활용하십시오.
- 하드웨어 디버깅 (하이브리드 워크플로우): FPGA 보드는 사용자의 로컬 머신(Windows 또는 macOS)에 연결되어 있을 것입니다. 이 경우, 원격 서버의 강력한 컴퓨팅 성능과 로컬 하드웨어 연결을 결합하는 하이브리드 워크플로우를 사용해야 합니다.
  1. FPGA 보드가 연결된 로컬 머신에서 Vivado Tcl 셸을 열고 `hw_server` 명령을 실행합니다. 이는 로컬 머신을 네트워크를 통해 접근 가능한 하드웨어 서버로 만듭니다.<sup>90</sup>
  2. 원격 서버에서 실행 중인 Vivado의 Hardware Manager에서 'Open New Target'을 선택합니다.
  3. 'Remote server' 옵션을 선택하고, 로컬 머신의 IP 주소를 입력하여 연결합니다.<sup>90</sup>

이 방식을 통해 원격 서버에서 생성된 비트스트림을 로컬 머신에 연결된 FPGA에 직접 프로그래밍하고, ILA(Integrated Logic Analyzer)를 이용한 디버깅을 수행할 수 있습니다.

### 11.3. 결론

Ubuntu 22.04 서버에 설치된 Vivado를 Windows 11 또는 macOS에서 원격으로 실행하기 위한 다양한 방법이 존재합니다. 각 기술은 고유한 장단점을 가지고 있지만, 종합적인 분석 결과, 경량 데스크톱 환경(XFCE)을 기반으로 적절하게 구성되고 보안이 강화된 RDP(xrdp) 연결이 가장 견고하고 성능이 뛰어난 솔루션을 제공합니다. 특히 인터넷과 같은 광역 네트워크(WAN)를 통해 작업해야 하는 현대의 분산된 개발 환경에서 RDP의 효율성은 다른 기술들을 압도합니다. 본 가이드에서 제시된 단계별 지침과 보안 모범 사례를 따르면, FPGA 엔지니어는 위치에 구애받지 않고 강력한 서버 리소스를 활용하여 생산적이고 안전한 원격 Vivado 개발 환경을 성공적으로 구축할 수 있을 것입니다.

### 참고 자료

1. VNC vs RDP: Which Remote Desktop Tool is the Best? - RealVNC, 9월 18, 2025에 액세스, <https://www.realvnc.com/en/blog/vnc-vs-rdp/>
2. XRDP (RDP) vs VNC - Which one to choose? - TheServerHost, 9월 18, 2025에 액세스, <https://theserverhost.com/blog/post/xrdp-vs-vnc>
3. Ask HN: Why is there no performant remote desktop for Mac/Linux? - Hacker News, 9월 18, 2025에 액세스, <https://news.ycombinator.com/item?id=32526108>
4. VNC vs RDP: which remote desktop solution to choose - NordLayer, 9월 18, 2025에 액세스, <https://nordlayer.com/blog/vnc-vs-rdp-key-differences/>
5. About Virtual Network Computing (VNC) - About Axis, 9월 18, 2025에 액세스, <https://docs.axissecurity.com/docs/about-vnc>
6. The shift away from X11 means no more forwarding X11 applications over ssh, wh... | Hacker News, 9월 18, 2025에 액세스,

- <https://news.ycombinator.com/item?id=24430825>
7. Unless RDP and VNC have fixed their performance problems They are both much, m... | Hacker News, 9월 18, 2025에 액세스, <https://news.ycombinator.com/item?id=8020408>
  8. openvpn - SSH X11 forwarding is extremely slow over VPN - Server ..., 9월 18, 2025에 액세스, <https://serverfault.com/questions/751535/ssh-x11-forwarding-is-extremely-slow-over-vpn>
  9. What You Need to Know About X11 Forwarding - Teleport, 9월 18, 2025에 액세스, <https://goteleport.com/blog/x11-forwarding/>
  10. X11 Forwarding: What Is It, Why Use It, How to Set It Up - StrongDM, 9월 18, 2025에 액세스, <https://www.strongdm.com/what-is/x11-forwarding>
  11. X11 Forwarding - Stanford Research Computing, 9월 18, 2025에 액세스, [https://stanford-rc.github.io/docs-earth/docs/x11\\_forwarding](https://stanford-rc.github.io/docs-earth/docs/x11_forwarding)
  12. Understanding X11 Forwarding through SSH - start to finish steps, 9월 18, 2025에 액세스, <https://superuser.com/questions/1405848/understanding-x11-forwarding-through-ssh-start-to-finish-steps>
  13. Should I use VNC or X protocol to control a Linux application from Windows? - Super User, 9월 18, 2025에 액세스, <https://superuser.com/questions/395119/should-i-use-vnc-or-x-protocol-to-control-a-linux-application-from-windows>
  14. All you need to know about VNC remote access technology - RealVNC blog, 9월 18, 2025에 액세스, <https://discover.realvnc.com/what-is-vnc-remote-access-technology>
  15. Virtual Network Computing (VNC) - How it works - GreenCloud, 9월 18, 2025에 액세스, <https://blog.greencloudvps.com/virtual-network-computing-vnc-how-it-works.php>
  16. VNC - Wikipedia, 9월 18, 2025에 액세스, <https://en.wikipedia.org/wiki/VNC>
  17. What is VNC? All You Need to Know About Remote Access Technology - RealVNC, 9월 18, 2025에 액세스, <https://www.realvnc.com/en/blog/what-is-vnc/>
  18. XRDP Installation and Configuration - Technical Support - NetSarang Computer, 9월 18, 2025에 액세스, <https://netsarang.atlassian.net/wiki/spaces/ENSUP/pages/2242871297/Ubuntu+22.04+-+XRDP+Installation+and+Configuration>
  19. Remote Server Access: RDP, SSH, or VNC? - WP Fastest Cache Premium, 9월 18, 2025에 액세스, <https://www.wpfastestcache.com/blog/remote-server-access-rdp-ssh-or-vnc/>
  20. X11 forwarding is insanely slow : r/linuxquestions - Reddit, 9월 18, 2025에 액세스, [https://www.reddit.com/r/linuxquestions/comments/r5kwbz/x11\\_forwarding\\_is\\_insanely\\_slow/](https://www.reddit.com/r/linuxquestions/comments/r5kwbz/x11_forwarding_is_insanely_slow/)
  21. Differences Between XRDP and VNC: Detailed Comparison - HelpWire, 9월 18, 2025에 액세스, <https://www.helpwire.app/blog/xrdp-vs-vnc/>
  22. Securing VNC with SSH - MIT, 9월 18, 2025에 액세스,

- <https://www.mit.edu/~avp/lqcd/ssh-vnc.html>
23. Remote Desktop (VNC) with SSH Tunnel, 9월 18, 2025에 액세스,  
<https://sscf.ucsd.edu/self-help/tutorials/linux/vnc-with-ssh-tunnel>
  24. How to connect to VNC using SSH - TechRepublic, 9월 18, 2025에 액세스,  
<https://www.techrepublic.com/article/how-to-connect-to-vnc-using-ssh/>
  25. How secure is open xrdp to internet? · Issue #702 - GitHub, 9월 18, 2025에 액세스,  
<https://github.com/neutrinolabs/xrdp/issues/702>
  26. XRDP – Configuring TLS connection for Remote Desktop Protocol – Griffon's IT Library, 9월 18, 2025에 액세스, <https://c-nergy.be/blog/?p=19845>
  27. How to Install and Configure SSH on Ubuntu 22.04 - Hostman, 9월 18, 2025에 액세스,  
<https://hostman.com/tutorials/how-to-install-and-configure-ssh-on-ubuntu-22-04/>
  28. OpenSSH server - Ubuntu documentation, 9월 18, 2025에 액세스,  
<https://documentation.ubuntu.com/server/how-to/security/openssh-server/>
  29. How to Enable SSH on Ubuntu 20.04 and 22.04 - GeeksforGeeks, 9월 18, 2025에 액세스,  
<https://www.geeksforgeeks.org/linux-unix/how-to-enable-ssh-on-ubuntu/>
  30. How to install a VNC server on Ubuntu 22.04 - IONOS, 9월 18, 2025에 액세스,  
<https://www.ionos.com/digitalguide/server/configuration/vnc-server-ubuntu-22-04/>
  31. Install and Configure VNC Server on Ubuntu 22.04|20.04|18.04 - ComputingForGeeks, 9월 18, 2025에 액세스,  
<https://computingforgeeks.com/how-to-install-vnc-server-on-ubuntu/>
  32. Enable Remote Desktop Protocol Using xrdp on Ubuntu 22.04 ..., 9월 18, 2025에 액세스,  
<https://www.digitalocean.com/community/tutorials/how-to-enable-remote-desktop-protocol-using-xrdp-on-ubuntu-22-04>
  33. How to Install and Configure VNC on Ubuntu 22.04 | DigitalOcean, 9월 18, 2025에 액세스,  
<https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-vnc-on-ubuntu-22-04>
  34. UFW Essentials: Common Firewall Rules and Commands for Linux ..., 9월 18, 2025에 액세스,  
<https://www.digitalocean.com/community/tutorials/ufr-essentials-common-firewall-rules-and-commands>
  35. How To Install and Config UFW Firewall on Ubuntu 22.04 - VPS Mart, 9월 18, 2025에 액세스,  
<https://www.vps-mart.com/blog/how-to-configure-firewall-with-ufw-on-ubuntu>
  36. Configure Ubuntu for X11 Forwarding on startup - Engineering Computer Labs & Laptops - UC Irvine, 9월 18, 2025에 액세스,  
<https://laptops.eng.uci.edu/engineering-software/using-linux/configure-ubuntu-for-x11-forwarding-on-startup>
  37. Enable X11 Forwarding | Nectar Research Cloud tutorials, 9월 18, 2025에 액세스,  
<https://tutorials.rc.nectar.org.au/x11forwarding/02-enable-x11-on-virtual-machine>

38. How to enable X11 forwarding from Red Hat Enterprise Linux ... - AWS, 9월 18, 2025에 액세스,  
<https://aws.amazon.com/blogs/compute/how-to-enable-x11-forwarding-from-red-hat-enterprise-linux-rhel-amazon-linux-suse-linux-ubuntu-server-to-support-gui-based-installations-from-amazon-ec2/>
39. How to Install VNC Server Ubuntu 22.04 - HowtoForge, 9월 18, 2025에 액세스,  
<https://www.howtoforge.com/how-to-install-vnc-server-ubuntu-22-04/>
40. VNC Server setup on Ubuntu 22.04 LTS - GitHub Gist, 9월 18, 2025에 액세스,  
<https://gist.github.com/indyfromoz/739cd53d47b91ba1d3b540ab53b1f46c>
41. Install & Configure VNC Server on Ubuntu 22.04 LTS - DEZRIN's LAB Wiki, 9월 18, 2025에 액세스, <https://wiki.dezr.in/en/install-configure-vnc-server-ubuntu-22>
42. VNC in Ubuntu 22.04 LTS, 9월 18, 2025에 액세스,  
<https://askubuntu.com/questions/1424361/vnc-in-ubuntu-22-04lts>
43. How to change to other desktop environment on VNC sessions? - Ask Ubuntu, 9월 18, 2025에 액세스,  
<https://askubuntu.com/questions/352232/how-to-change-to-other-desktop-environment-on-vnc-sessions>
44. Installing Xfce and XRDP Service on Remote Ubuntu 22.04 - VPS Mart, 9월 18, 2025에 액세스,  
<https://www.vps-mart.com/blog/install-xfce-and-xrdp-service-on-remote-ubuntu>
45. How to configure XRDP to work with Gnome on Ubuntu? [closed] - Stack Overflow, 9월 18, 2025에 액세스,  
<https://stackoverflow.com/questions/78074498/how-to-configure-xrdp-to-work-with-gnome-on-ubuntu>
46. xrdp failing on Ubuntu 22.04 with remmina client · Issue #3499 - GitHub, 9월 18, 2025에 액세스, <https://github.com/neutrinolabs/xrdp/issues/3499>
47. Set Up xrdp on Ubuntu 22.04 With the Gnome Desktop Environment - Technical Scratchpad, 9월 18, 2025에 액세스,  
<https://luppeng.wordpress.com/2024/03/12/set-up-xrdp-on-ubuntu-22-04-with-the-gnome-desktop-environment/>
48. Remote Desktop from Windows onto Ubuntu 22.04 takes me to a XRDP login then a blank screen, 9월 18, 2025에 액세스,  
<https://askubuntu.com/questions/1404245/remote-desktop-from-windows-onto-ubuntu-22-04-takes-me-to-a-xrdp-login-then-a-bl>
49. How to Access Ubuntu via Remote Desktop from Windows, 9월 18, 2025에 액세스, <https://phoenixnap.com/kb/ubuntu-remote-desktop-from-windows>
50. Remote Desktop connection from Mac to Ubuntu, 9월 18, 2025에 액세스,  
<https://askubuntu.com/questions/893831/remote-desktop-connection-from-mac-to-ubuntu>
51. How to use X11 forwarding with PuTTY - Super User, 9월 18, 2025에 액세스,  
<https://superuser.com/questions/119792/how-to-use-x11-forwarding-with-putty>
52. Run an X11 Application on Windows | Information Technology and ..., 9월 18, 2025에 액세스,  
<https://it.engineering.oregonstate.edu/run-x11-application-windows>



53. MobaXterm Xserver with SSH, telnet, RDP, VNC and X11 - Features - Mobatek, 9월 18, 2025에 액세스, <https://mobaxterm.mobatek.net/features.html>
54. MobaXterm Xserver with SSH, telnet, RDP, VNC and X11 - Home Edition - Mobatek, 9월 18, 2025에 액세스, <https://mobaxterm.mobatek.net/download-home-edition.html>
55. Running Graphical Linux Programs Remotely (X Forwarding) - Service Desk, 9월 18, 2025에 액세스, <https://servicedesk.mtu.edu/TDClient/1801/Portal/KB/ArticleDet?ID=103526>
56. X11 Forwarding - UW Hyak, 9월 18, 2025에 액세스, <https://hyak.uw.edu/docs/setup/x11/>
57. X-Forwarding on Windows - Lehigh Computer Science Docs, 9월 18, 2025에 액세스, <https://docs.cse.lehigh.edu/xforwarding/xforwarding-win/>
58. PuTTY/Xming - CAC TechDocs - CAC Portal, 9월 18, 2025에 액세스, <https://portal.cac.cornell.edu/techdocs/clusterinfo/puttyxming/>
59. How to Use X11 Forwarding on Windows or Linux - YouTube, 9월 18, 2025에 액세스, [https://www.youtube.com/watch?v=FIHVuA\\_98SA](https://www.youtube.com/watch?v=FIHVuA_98SA)
60. How do I get X11 Forwarding to work on Windows with PuTTY and Xming? - Super User, 9월 18, 2025에 액세스, <https://superuser.com/questions/592185/how-do-i-get-x11-forwarding-to-work-o-n-windows-with-putty-and-xming>
61. VNC Instructions for Windows - CSUSB, 9월 18, 2025에 액세스, [https://www.csusb.edu/sites/default/files/cse\\_vnc\\_windows.pdf](https://www.csusb.edu/sites/default/files/cse_vnc_windows.pdf)
62. How to Install Tiger VNC on Windows 11 - IPv6rs, 9월 18, 2025에 액세스, [https://ipv6.rs/tutorial/Windows\\_11/Tiger\\_VNC/](https://ipv6.rs/tutorial/Windows_11/Tiger_VNC/)
63. TigerVNC, 9월 18, 2025에 액세스, <https://tigervnc.org/>
64. Installing and configuring VNC client to connect through an SSH tunnel - TeamDynamix, 9월 18, 2025에 액세스, <https://teamdynamix.umich.edu/TDClient/47/LSAPortal/KB/PrintArticle?ID=4857>
65. Enable Remote Desktop on your PC - Microsoft Learn, 9월 18, 2025에 액세스, <https://learn.microsoft.com/en-us/windows-server/remote/remote-desktop-services/remotepc/remote-desktop-allow-access>
66. Connect to Ubuntu Remotely from Windows 11 using RDP/XRDP - YouTube, 9월 18, 2025에 액세스, <https://www.youtube.com/watch?v=kkn3gQUnxzk>
67. Mailing List FAQs - XQuartz, 9월 18, 2025에 액세스, <https://www.xquartz.org/FAQs.html>
68. XQuartz, 9월 18, 2025에 액세스, <https://www.xquartz.org/>
69. How to do X11 Forwarding on a Mac - ECI Public Knowledgebase - Confluence, 9월 18, 2025에 액세스, <https://ucsb-engr.atlassian.net/wiki/spaces/EPK/pages/573603972/How+to+do+X11+Forwarding+on+a+Mac>
70. X-Forwarding on Mac OS - Lehigh Computer Science Docs, 9월 18, 2025에 액세스, <https://docs.cse.lehigh.edu/xforwarding/xforwarding-mac/>
71. SSH with X Forwarding : TechWeb - Boston University, 9월 18, 2025에 액세스, <https://www.bu.edu/tech/support/research/system-usage/connect-scc/ssh-x-forwarding/>

72. Remote Access Setup for MacOS (SSH + VNC), 9월 18, 2025에 액세스,  
[https://wcsng.ucsd.edu/docs/technical/remote\\_access/remote\\_access\\_macos/](https://wcsng.ucsd.edu/docs/technical/remote_access/remote_access_macos/)
73. VNC from Mac to Linux - Computer Action Team - Portland State University, 9월 18, 2025에 액세스,  
<https://cat.pdx.edu/platforms/mac/remote-access/vnc-to-linux/>
74. Secure VNC screen sharing connection from Mac (client) to Windows10 (server) via SSH Tunnel - Super User, 9월 18, 2025에 액세스,  
<https://superuser.com/questions/1870276/secure-vnc-screen-sharing-connection-from-mac-client-to-windows10-server-via>
75. Tunneling VNC Through SSH on macOS | Khalid Abuhakmeh, 9월 18, 2025에 액세스,  
<https://khalidabuhakmeh.com/tunneling-vnc-through-ssh-on-macos>
76. Remote Desktop Connection from Mac to Ubuntu - phoenixNAP, 9월 18, 2025에 액세스,  
<https://phoenixnap.com/kb/ubuntu-remote-desktop-from-mac>
77. How to Access Remote Ubuntu Desktop from a Mac - Tecmint, 9월 18, 2025에 액세스,  
<https://www.tecmint.com/connect-ubuntu-desktop-from-mac/>
78. What does Vivado need to perform as best/fast as possible? : r/FPGA - Reddit, 9월 18, 2025에 액세스,  
[https://www.reddit.com/r/FPGA/comments/1bme4k3/what\\_does\\_vivado\\_need\\_to\\_perform\\_as\\_bestfast\\_as/](https://www.reddit.com/r/FPGA/comments/1bme4k3/what_does_vivado_need_to_perform_as_bestfast_as/)
79. How to launch and automatically feed commands to a software console? - Stack Overflow, 9월 18, 2025에 액세스,  
<https://stackoverflow.com/questions/77744657/how-to-launch-and-automatically-feed-commands-to-a-software-console>
80. Running VIVADO project from batch- linux , by using tcl file. : r/FPGA - Reddit, 9월 18, 2025에 액세스,  
[https://www.reddit.com/r/FPGA/comments/11veudi/running\\_vivado\\_project\\_from\\_batch\\_linux\\_by\\_using/](https://www.reddit.com/r/FPGA/comments/11veudi/running_vivado_project_from_batch_linux_by_using/)
81. Vivado Design Suite Tcl Command Reference Guide, 9월 18, 2025에 액세스,  
[https://www.xilinx.com/support/documents/sw\\_manuals/xilinx2022\\_1/ug835-vivado-tcl-commands.pdf](https://www.xilinx.com/support/documents/sw_manuals/xilinx2022_1/ug835-vivado-tcl-commands.pdf)
82. xorg - X11 - Forwarding and efficiency - Super User, 9월 18, 2025에 액세스,  
<https://superuser.com/questions/1400952/x11-forwarding-and-efficiency>
83. Using X11 forwarding painfully slow : r/termux - Reddit, 9월 18, 2025에 액세스,  
[https://www.reddit.com/r/termux/comments/sgiltc/using\\_x11\\_forwarding\\_painfully\\_slow/](https://www.reddit.com/r/termux/comments/sgiltc/using_x11_forwarding_painfully_slow/)
84. networking - How can I make VNC faster? - Super User, 9월 18, 2025에 액세스,  
<https://superuser.com/questions/15337/how-can-i-make-vnc-faster>
85. Using vnc to do remote tech support over high-latency networks - Feeding the Cloud, 9월 18, 2025에 액세스,  
<https://feeding.cloud.geek.nz/posts/high-latency-vnc-tech-support/>
86. How can I optimize the use of TightVNC? | IDEAL Administration : Simplify Windows Active Directory and Workgroup Management with a Single Console | Pointdev, 9월 18, 2025에 액세스,  
<https://www.pointdev.com/en/faq/faq-ideal-administration-optimize-tightvnc-vnc-ultravnc-id-230.html>



87. How to Set Up SSH Keys on Ubuntu: A Comprehensive Guide ..., 9월 18, 2025에 액세스,  
<https://www.digitalocean.com/community/tutorials/how-to-set-up-ssh-keys-on-ubuntu-22-04>
88. Installing and Configuring OpenSSH Server on Ubuntu 22 - Reintech, 9월 18, 2025에 액세스,  
<https://reintech.io/blog/installing-configuring-openssh-server-ubuntu-22>
89. Blocking IP or Whitelisting IP Addresses with UFW | Liquid Web, 9월 18, 2025에 액세스,  
<https://www.liquidweb.com/blog/blocking-ip-or-whitelisting-ip-addresses-with-u fw/>
90. Using Vivado Remotely - Hackster.io, 9월 18, 2025에 액세스,  
<https://www.hackster.io/elliott-koehn/using-vivado-remotely-0e302f>