GoPhish Phishing Simulation

Objective

I set up and ran a simulated phishing campaign using GoPhish in a VM lab to see how a realistic phishing email and landing page perform against test accounts. The goal was to demonstrate what a credential-harvesting attack looks like and to collect simple metrics (delivered, opened, clicked, credentials submitted).

Environment & tools

I ran GoPhish on a Kali Linux VM and used a separate test inbox on the isolated lab network to receive messages. Tools: GoPhish web UI, a browser (for testing and viewing results), and the VM screenshot tool to capture evidence.

What I did

1. Launched GoPhish and created a sending profile with SMTP settings, then sent a test message to confirm delivery.
2. Built a landing page that mimics a login form and set it to redirect to a legitimate page after submission (this keeps the simulation realistic and harmless).
3. Wrote a short email template that asks the recipient to "sign in" to view an important message and embedded the landing page link.
4. Added a small set of test accounts (lab/consenting accounts only) as targets and launched the campaign.
5. Monitored the GoPhish dashboard as the campaign progressed and took screenshots of the dashboard, the received email in the test inbox, and the landing page.
6. When a test user submitted credentials on the landing page, GoPhish recorded them in the campaign results.

What happened

The SMTP test succeeded and the campaign was delivered to the test inbox.

- The dashboard updated in real time showing which messages were delivered and which recipients opened the email and clicked the link.
- At least one test account clicked the link and submitted credentials on the landing page. GoPhish captured those submissions for analysis.
- The whole flow — send → open → click → submit — was quick and clearly visible in the GoPhish interface.

Findings

- A realistic-looking email plus a convincing landing page is effective in a lab environment. Even simple templates can trick a test user into submitting credentials.

- GoPhish provides an easy way to measure user susceptibility (opens, clicks, submissions) and collect evidence for training.
- The main risk demonstrated is credential harvesting: if users reuse passwords, attackers can use captured credentials to access real accounts.

Impact

If this were a real organization without protections like MFA and good user training, attackers could obtain credentials and potentially access sensitive systems.