GREEN SHELL TECHNOLOGY

"Digital architecture done right"

What is Greenshell? We're an IT company, specializing in delivering computing solutions for independently-minded home and small business owners.

Background; the history of computing:

1960s - 1980s: Mainframes
1980s - 2000: Personal computer
2000 - 2020: Cloud computing
2020 - ?: Self-hosted solutions

Why doesn't this exist yet?

I. Problems with traditional, "on-prem" solutions. Cloud is simply ***dominant***; better service and products all around.
II. Cracks opening up:
    A. Security; ransomware nightmare
    B. Control; moderation and censorship
    C. Expense and waste; planned obsolescence, huge fees, increasingly invisible benefit
    D. Technical know-how
III. Today, in 2021, the tools exist to make a self-hosted, distributed, small-scale system accessible and operable by independent operators. See homelab.pdf

What we are *not*

I. A managed service provider. No SLAs, no guarantees of service, no remote operation & maintenance of client networks.
    A. Firstly, we're a 1) reseller of existing, high-quality products (Qnap for NAS, Microtik for routing, Foscam for cameras, etc)
    B. Secondly, we do 2) initial, physical network installation. You need to hire someone to actually set up cameras, run wires, secure your network, configure everything correctly, etc.
    C. Thirdly, and most importantly, we provide education, training, and *SOME* level of informal support, well below the standards of SLAs
II. A hardware, firmware, or software company. *Yet*.

A. For the foreseeable future, Greenshell does not produce any technology itself. We can build a sustainable, profitable business using off-the-shelf products & latest-and-greatest best practices. Merely adding value in the above categories, and enabling and training people to be self-reliant sysadmins in their own right.

The architecture itself

Here's where the fun begins :). Several layers:

1) WAN connection, modem, and provided router will be provided by your local ISP
2) Link layer; physical ethernet ran throughout your house in a tidy, professional, and maintainable manner. Perhaps we also provide WiFi support, relays, etc., (but not an essential part of the business; merely a profitable add-on)
3) Devices;
    a) Consumer devices, like iPhone, Macbook, or Windows computer will be taken as-is, whatever the customer prefers
    b) Cameras for surveillance; we select the best product, make sure it works with our architecture, and provide "light" support for any firmware upgrades or other maintenance issues. **User must deploy firmware upgrades themselves, or pay for limited, one-time support services as they arise. *NO* contracts.**
    c) Routers; we select the best product, makes sure it works with architecture, etc. See above.
    d) NAS devices; likely the most active area of development and support. Architecture should support the full range of reasonable reliability, from "plug-and-forget" single-device installation, to multiple offsite backups (either in another client-accessible location, or in a **local** "datacenter" operated by us [See note on this later]).
        i) This also includes selecting & replacing the proper type of hard drive for the customer's use case.
    e) Any application servers needed to run any supported solutions.
        i) For initial use cases, a single inexpensive, commodity i7 tower should do the trick for everything we need.
        ii) Obviously, as user needs grow, the compute requirements for the whole network grow as well.
        iii) Unlikely to support any kind of "big compute" use case; machine learning, complex rendering, applications with 1000+ users, or anything that takes a ginormous amount of compute or storage space. Just use the cloud for that :).
4) Networking
    a) On an actual client network, our self-hosted solutions should be an isolated, self-contained loop. Essentially; no outside access for any reason. No public IP or services.
    b) Second layer is providing a roll-your-own VPN and encryption services, likely using Wireguard and LetsEncrypt respectively. This will be used for things like:
        i) Remote monitoring of surveillance setup

ii) Remote access to files, media, or other types of application data you need on the fly.
iii) Limited access to a small-scale communication platform for trusted, IRL friends & family

5) Finally; the applications themselves. Here's where all the true fun is in the long-term.
   a) Initially, limited to the specific solutions we offer; surveillance, data backup, etc.
      i) Surveillance: http://kerberos.io
      ii) Data backup: https://syncthing.net/
      iii) Media hosting & distribution: {Plex?}
      iv) Social networking: http://matrix.org
      v) Other fantastic software solutions, suites, etc:
         (1) https://nextcloud.com/
         (2) https://github.com/awesome-selfhosted/awesome-selfhosted
         (3) Sky is truly the limit. Huge list of things here to evaluate, install, contribute to, and provide light support on.
   b) Eventually, we hope to expand our range to include all of the best-available and growing self-hosted solutions (FOSS or not)
   c) Big end goal & final payoff is developing a full-fledged "app store" for self-hosted projects, which works natively and beautifully with the rest of the architecture we provide.

Q: How do we get there?
A: Strategy and tactics

Strategy

I. **Build a profit-generating business doing low-volt installation for homes and small businesses.**
   A. Some initial investment may be required. But the goal is to run like a traditional business; no hypergrowth, just delivering a better product & service than existing solutions.
   B. Grow the business in a traditional way; hiring people, satisfying clients,

II. **Find product-market fit.** Places to start include:
   A. Surveillance systems
   B. "Personal cloud", storage and backup
   C. "Personal cloud", media hosting & distribution
   D. Small-scale, semi-private social networks.
   E. Become an "app store" for other self-hosted solutions (https://github.com/awesome-selfhosted/)

III. **Hone the architecture**
   A. An IT solution has to eventually excel at the following properties, in various orders:
      1. Performance
      2. Reliability

3. Availability
4. Security
5. Privacy
6. Finally, scaling

B. Initially, we use off-the-shelf, reasonably standard sets of hardware, firmware, and software. FOSS-driven by necessity.
C. Once we get rolling as a business, and have more money to play with, we can invest more in R&D, in tandem with the needs of our clients and the movements of the market.

IV. Be competitively nimble; engage in business strategy

A. Eventually, when we reach a certain size (??), we may find that there are bottlenecks with suppliers (at any layer; software/firmware/hardware). We should be always vigilant and ready to invest in our own custom-made solutions wherever necessary and wherever the market dictates.
B. Hopefully will not have to worry about this for a long while.
C. Will rely on networks of other companies, individuals, and investors with similar values and principles. Work with other companies and architectures where possible.

Tactics

I. Tactics -- business
   A. Clients
      1. So far, user feedback is incredibly positive. People seem to understand the value proposition & want it.
      2. Marketing is very ripe, but strategic balance must be maintained at all steps. Don't oversell what you're offering, and focus only on limited value propositions at first. Just compete with existing products/services and do an objectively better job than they do. Leave clients satisfied.
      3. Choose clients carefully.
   B. Partners
      1. Hire the right people.
      2. Currently, there's no funding, and no cashflow. Equity-invested partners that believe in the vision, can do the job, and can sustain themselves are a must.
      3. A *small* amount of personal funding can and will be allocated to things like limited design contracts for marketing purposes.
   C. Funding
      1. Find the right investors.
   D. Profit
      1. All of this stuff needs to eventually shake out in a way that the numbers support the business.
      2. Track everything, collect data on everything, and optimize at every step of the way. Initially, things will be slow and expensive, but will improve over time.

3. *Never* compromise on quality; leave every customer, partner, and investor fully satisfied.
4. Once you get rolling, forget about the big picture, and just execute. The big picture will always be there. Try to spend as little time as possible on “general business development”. Advertise and market only as much as you need to keep the client list full & cash flow positive. Avoid unnecessary attention.

II. Tactics -- development
   A. Learn how the hell any of this IT stuff actually works, haha. (Or hire someone who does).
   B. Prototype out a system that delivers on the core use case, surveillance
   C. Get out there and install an actual, working system.