Freelance Data / DevOps / Fullstack / Linux engineer

|  |  |
| --- | --- |
| name | ing. Sander Lentink, MSc. |
| currentcity | Lelystad |
| phone | +31617012655 |
| email | sv@lent.ink |
| links | github.com/svlentink blog.lent.ink |

## About

Eager to create and learn, diligent and always looking to get the job done. Technologies are solely a means to get the job done. Security minded and willing to discreetly address issues.

### Ambition

Love to deepen my experience as a data (platform) engineer and like it when my work incorporates data science, security and writing code.

# Education

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MSc. | Security and Network engineering | University of Amsterdam | 2017-2019 | &#x1f393; |
| BSc. | Software engineering | Windesheim | 2012-2016 | &#x1f393; |
| Propedeuse | Computer engineering | Windesheim | 2011-2012 | &#x1f393; |
| community college | Mechatronics | Landstede | 2007-2011 | &#x1f393; |

## Certificates and courses

|  |  |  |  |
| --- | --- | --- | --- |
| minor | Artificial Intelligence | University of Amsterdam | 2014 |
| minor | Mathematics | Utrecht University | 2014 |
| minor | Web technology | Windesheim | 2013 |
| cert. | 7 habits of highly effective people | FranklinCovey | November 2012 |
| cert. | Introduction Android Development | InfoSupport | December 2011 |
| cert. | Basic Safety VCA | Cito | February 2007 |

# Knowledge fields

knowledge tree   
 +-- Configuration Management CM   
 | +-- CI/CD integration, deployment   
 | | +-- Azure DevOps November 2019 -> 2023   
 | | +-- Gitlab August 2018 -> November 2018   
 | | +-- Travis February 2017 -> October 2018   
 | +-- Infrastructure as Code IaC   
 | | +-- Ansible March 2017 -> August 2019   
 | | +-- Terraform November 2018 -> 2023   
 | +-- Version Control Systems VCS   
 | +-- Git 2013 -> 2023   
 | | +-- Gitlab   
 | | +-- Github   
 | | +-- Bitbucket   
 | +-- svn 2012 -> 2013   
 +-- containers   
 | +-- Docker February 2015 -> 2023   
 | +-- LXC March 2017 -> 2018   
 | | +-- LXD November 2017 -> 2018   
 | +-- Rkt November 2017 -> December 2017   
 | +-- container orchestration   
 | +-- DC/OS Mesosphere February 2017 -> November 2018   
 | +-- Kubernetes June 2018 -> 2023   
 | +-- Microservices June 2016 -> 2018   
 +-- data   
 | +-- cache   
 | | +-- Memcache April 2017 -> October 2018   
 | | +-- Redis June 2016 -> October 2018   
 | +-- data science   
 | | +-- D3js 2015 -> April 2018   
 | | +-- Python   
 | | | +-- Numpy September 2018 -> October 2018   
 | | | +-- Pandas September 2018 -> November 2018   
 | | +-- web scraping   
 | +-- database   
 | | +-- MySQL / MariaDB October 2011 -> 2023   
 | | +-- NoSQL   
 | | | +-- MongoDB July 2016 -> October 2018   
 | | | +-- OrientDB March 2015 -> June 2015   
 | | +-- PostgreSQL   
 | | +-- SQL server 2012 -> 2013   
 | +-- formats   
 | +-- AMQP / RabbitMQ February 2017 -> 2023   
 | +-- JSON   
 | +-- XML   
 | +-- YAML   
 +-- infrastructure   
 | +-- FreeRADIUS October 2019 -> December 2019   
 | +-- Linux   
 | | +-- Alpine Linux 2016 -> 2023   
 | | +-- Redhat April 2019 -> August 2019   
 | | +-- Ubuntu / Debian 2012 -> 2023   
 | +-- OpenStack January 2017 -> February 2017   
 | +-- VM   
 | | +-- KVM February 2017 -> October 2018   
 | | +-- Xen November 2017 -> 2023   
 | +-- cloud   
 | | +-- AWS Amazon Web Services February 2017 -> November 2018   
 | | +-- DNS 2016 -> 2023   
 | | | +-- bind September 2018 -> October 2018   
 | | +-- Load balancing July 2017 -> September 2018   
 | | | +-- HA-proxy March 2016 -> October 2018   
 | | +-- Microsoft Azure June 2019 -> June 2020   
 | | +-- Reverse / TLS proxy   
 | | +-- Apache January 2017 -> September 2018   
 | | +-- Nginx 2015 -> 2023   
 | +-- monitoring and alerting   
 | +-- Elastic Search April 2017 -> July 2018   
 | | +-- Kibana   
 | | +-- Logstash / grok November 2017 -> December 2017   
 | +-- Email   
 | | +-- Exim April 2017 -> 2019   
 | | +-- smarthost   
 | +-- Nagios December 2016 -> November 2018   
 | +-- Slack API 2017 -> 2018   
 +-- languages   
 | +-- Bash February 2015 -> 2023   
 | +-- C# January 2013 -> June 2013   
 | +-- CSS3 October 2011 -> 2023   
 | +-- Go (golang) June 2019 -> September 2019   
 | +-- HTML5 September 2011 -> 2023   
 | +-- Java April 2012 -> January 2014   
 | +-- Javascript February 2015 -> 2023   
 | +-- LaTeX 2014 -> 2019   
 | +-- Node.js February 2015 -> 2018   
 | +-- PHP September 2011 -> June 2012   
 | +-- Python February 2017 -> 2023   
 | +-- R November 2017 -> March 2018   
 | +-- Ruby February 2016 -> May 2016   
 | +-- SQL September 2011 -> 2023   
 +-- process management   
 | +-- SDLC   
 | | +-- Agile   
 | | | +-- Kanban 2013 -> 2023   
 | | | +-- Scrum April 2013 -> 2023   
 | | | +-- User stories   
 | | +-- Waterfall October 2011 -> 2012   
 | +-- documentation   
 | | +-- Confluence April 2017 -> October 2019   
 | | +-- Dokuwiki 2017 -> 2019   
 | | +-- Markdown   
 | | +-- draw.io   
 | +-- ticketing system   
 | +-- Github issue system 2014 -> 2023   
 | +-- Jira 2016 -> 2019   
 | +-- Redmine April 2013 -> June 2013   
 | +-- Trello 2013 -> 2018   
 +-- security   
 +-- Access control   
 | +-- Basic-AUTH May 2017 -> 2023   
 | +-- IPTables May 2017 -> 2019   
 | +-- Netfilter   
 | +-- VPN 2017 -> 2023   
 +-- Cryptography   
 | +-- Elliptic-curve cryptography (ECC)   
 | +-- Public Key Infrastructure (PKI)   
 | +-- RSA   
 | +-- Secure Socket Layer (SSL)   
 | +-- Transport Layer Security (TLS)   
 +-- DNSSEC October 2018 -> 2019   
 +-- Kali linux February 2018 -> April 2018   
 +-- Snort April 2018 -> April 2018   
 +-- Wireshark / tcpdump 2014 -> 2019

## Other

AWS S3, Arduino, Chrome extension, DRY Don't repeat yourself, Flask, FreeRADIUS, Hugo, IPv6, IaaS, Internet of Things IoT, JQuery, KISS Keep it simple, Model View Controller (MVC), NFS, OpenSSH, OpenSSL, PaaS, Progressive Web App (PWA), REST API, RFC, SOAP, SSH, SaaS, Unified Modeling Language (UML), Wordpress, back-end, blockchain, data-mining, gcc, integration testing, makefile, routing, unit test, webpack

# Experience

|  |  |
| --- | --- |
| role | Senior Data engineer | Vattenfall (energy), Amsterdam |
| timespan | July 2021 -> June 2022, 40h p/w (consultant) |
| desc | Optimize offshore wind turbine maintenance by empowering data analysts with tooling, infrastructure and data, so their predictive models can identify the wear on turbine components. |
| activities | Translate business questions into complex SQL queries Creating designs for data extraction, transformation and loading (ETL) Configuring JWT authentication using Envoy proxy for web application backend Enabling developers to run VS-code on k8s Managing Cloud infrastructure through Infrastructure as Code (IaC) Enabling GitOps of AKS and DB on Azure Cloud using Terraform Data transformation in Python using Pandas |
| techniques | Scrum Python Pandas Envoy SQL Apache NiFi AKS Kubernetes Kustomize Dockerfile Linux Git Kafka Swagger PostgreSQL Azure DevOps Terraform Bash REST API PostgREST OmniDB MS Visio Azure Blob Nginx Nexus |

|  |  |
| --- | --- |
| role | Data Platform engineer | Tennet (energy), Arnhem |
| timespan | July 2020 -> June 2021, 40h p/w (consultant) |
| desc | Enabled the upscaling of renewable energy sources through the creation of data ingestion pipelines to facilitate data scientist in making predictive models. We developed services that ingested from various sources and wrote to HDFS or inserted into a database, while monitoring them using dashboards and alerting. |
| activities | Improving and stimulating innovation relating to security processes through standardization and automation Developing ingestion services using Python Deploying various services using GitOps to container orchestration platform Enabling GitOps of LDAP enabled Grafana instances Enabling developers to run VS-code on the DTAP container platforms Developing multi threaded Python service to ingest data using NiFi Enabling GitOps of Databases (DaaS) Managing prerequisites ingestion services; ACLs, service user, HDFS, Kafka Creation of dashboards visualizing ingestion metrics using Kibana Configuration of Filebeat on container orchestration platform Developing python package to implement standardized JSON log format Enabling GitOps of ElasticSearch watchers for alerting to Slack and email Creating and maintaining internal Python (pypi) packages Developing streaming API in Python between HDFS and HTTP |
| techniques | Scrum Python Dockerfile Linux CI/CD Git SQL Apache NiFi DC/OS Kafka Impala Hue Hadoop HDFS Oracle Swagger PostgreSQL Jenkins Mesos Marathon ELK ElasticSearch Logstash Kibana Grafana Filebeat Kerberos Bash PowerShell SOAP Webdav REST API SFTP Nexus |

|  |  |
| --- | --- |
| role | Cloud Architect | Rabobank (banking), Utrecht |
| timespan | November 2019 -> June 2020, 40h p/w (consultant) |
| desc | As part of the strategic move away from own data centers (DC) to the Cloud, I helped with setting up the infrastructure for the pilot application (Pega) by creating the automated rollout of a kubernetes (k8s) cluster including the application. We were responsible for designing and executing the first migration, in which I directed the implementation of Infrastructure as Code (IaC) using Terraform. |
| activities | Constant evaluation of business value vs. engineer's wishes for technical improvements Design of Infrastructure as Code (IaC) environment Integrate security principles in IaC, k8s and processes Develop standards and modules in Terraform Optimize CI/CD pipelines and its templates Automate deployment on Azure using Terraform and pipelines Convert existing installation guide to IaC Perform Database tests as part of selection process Direct the IaC team Assisting team with Terraform and container questions |
| techniques | Scrum Scaled Agile Framework (SAFe) Kubernetes Terraform Azure DevOps Python Dockerfile Linux CI/CD Git PostgreSQL Azure Pega (Java) Infrastructure as Code (IaC) |

|  |  |
| --- | --- |
| role | Network engineer | Surfnet (research), Utrecht |
| timespan | October 2019 -> November 2019, 40h p/w (thesis) |
| projecturl | http://tunroam.org |
| desc | Created wireless protocol that allows the creating of secure public WiFi access points by whitelisting VPN traffic. |
| activities | Modifying 802.1x authentication server to support protocol Design wireless protocol Implement protocol on a Raspberry Pi, making it a wireless Access Point |
| techniques | draw.io LaTeX FreeRADIUS Python Markdown Javascript Git Debian Linux Docker Bash Raspberry Pi Orange Pi Armbian OpenVPN |

|  |  |
| --- | --- |
| role | DevOps engineer | Rabobank (banking), Utrecht |
| timespan | March 2019 -> October 2019, 24h p/w (consultant) |
| desc | As part of the strategic move away from own data centers (DC) to the Cloud, I rationalized the on premise infrastructure. For this I used Infrastructure as Code (IaC) principles to create a manageable middleware to request resources in their private DC. |
| activities | Strategize transition to containers, to facilitate transition to the Cloud Research options to request resources via ManageIQ using Infrastructure as Code (IaC) Creating ManageIQ Terraform provider using Golang Develop new readable Terraform modules following AWS syntax Develop data parsers in Python for on premise DC management Create coupling between DevOps tools and APIs Optimizing existing IaC templates (YAML and JSON) Containerize Pega using Dockerfile (creation of golden image) Creating platform design drawings Assisting team with Python, Linux, containers, routing and Terraform questions Presenting insights and designs to teams |
| techniques | Scrum Scaled Agile Framework (SAFe) Terraform Docker Python Ansible Alpine Linux ManageIQ Git Go (golang) Swagger Redhat Bitbucket Confluence Infrastructure as Code (IaC) |

|  |  |
| --- | --- |
| role | DevOps Engineer | Mijndomein (IT), Lelystad |
| timespan | December 2016 -> November 2018, 24h p/w (employee) |
| desc | Automating operations related tasks and implementing new services. Maintaining the uptime of current services; hosting, email, DNS and upgrade internal configuration tools. |
| activities | Developing micro services using Python Enable and connect services to the RabbitMQ message system Optimize operations using Nagios, Elastic search monitoring, Slack notifications Maintenance hosting infrastructure Security audit the infrastructure Ddos intervention Spam and phising email monitoring and prevention Provide software developers with docker platform On-call shift Email platform maintenance Maintenance in data center |
| techniques | Scrum Kanban Python Apache LXC SQL Docker Kubernetes DC/OS Mesosphere Git Exim KVM Debian Linux AWS EC2 RabbitMQ PowerDNS HA-proxy OpenStack Kibana Dokuwiki CertManager |

|  |  |
| --- | --- |
| role | Covert channel researcher | OS3 (education), Amsterdam |
| timespan | April 2018 -> May 2018, 20h p/w (edu. project) |
| desc | Desk research into the possibilities of hiding data in pictures and sharing them on social media. Various social media converted the images (JPG and BMP), destroying the hidden data. |
| activities | Embed data using various stenography tools Contribute to an open source stenography tool on Github Uploading various data containing images and downloading for validation |
| techniques | Bash Imagemagick LaTeX Docker Git |

|  |  |
| --- | --- |
| role | Blockchain developer | OS3 (education), Amsterdam |
| timespan | February 2018 -> March 2018, 24h p/w (edu. project) |
| projecturl | http://github.com/svlentink/logdag |
| desc | Classic infrastructures use a central logging server. We created a Proof of Concept which distributes log chunks randomly to other servers and adds a hash of it a blockchain. This makes it infeasible to change the logs. |
| activities | Design and implement blockchain architecture Developing a web interface which shows a representation of the graph in real time Writing the backend code in Python |
| techniques | Python Docker HTML5 Javascript LaTeX D3js Git Dokuwiki |

|  |  |
| --- | --- |
| role | Blockchain researcher | KPMG (education), Amstelveen |
| timespan | January 2018 -> January 2018, 40h p/w (thesis) |
| desc | Most blockchain technologies demand a participant to always be online, in sync with the blockchain. We wrote a paper on the various consensus mechanisms from the blockchain landscape and showed which are applicable on mobile devices. |
| techniques | LaTeX Docker Git |

|  |  |
| --- | --- |
| role | Container technologist | OS3 (education), Amsterdam |
| timespan | November 2017 -> December 2017, 24h p/w (edu. project) |
| projecturl | https://github.com/svlentink/container-performance |
| desc | One of the new innovations is serverless. Serverless often requires you to rewrite code. We developed an alternative Proof of Concept, which when triggered (REST API, AMQP message) starts a container and returns its output. |
| activities | Develop PoC in python Generate performance overview using R Bootstrapping VPSes using Bash |
| techniques | Rkt Docker LXC / LXD Python Flask Bash LaTeX R Git |

|  |  |
| --- | --- |
| role | IoT Security researcher | OS3 (education), Amsterdam |
| timespan | September 2017 -> October 2017, 24h p/w (edu. project) |
| desc | Analysing the data flow of IoT (Internet of Things) devices, to get insight into the security and privacy. We looked at multiple devices, looking at the data and to where it connected to. |
| activities | Configuration IoT devices Analyse tcpdump using Wireshark |
| techniques | Wireshark / tcpdump LaTeX Docker Bash Git Dokuwiki |

|  |  |
| --- | --- |
| role | DevOps Engineer | Superbuddy (IT), Zwolle |
| timespan | June 2016 -> September 2016, 40h p/w (thesis) |
| desc | As part of my thesis, I've dockerized their monolith and enabled AB testing by separating components into microservices. Adding better monitoring and deployment via CI/CD. |
| activities | System recommendations Dockerizing existing applications Implement CI/CD Enable A/B testing using nginx Advise on VPS providers |
| techniques | Scrum Git Bash Docker Docker-compose Nginx PHP draw.io LaTeX Gitlab |

|  |  |
| --- | --- |
| role | System analyst | Blendle (digital journalism), Utrecht |
| timespan | February 2016 -> May 2016, 40h p/w (thesis) |
| desc | To reduce the loading time of the application, I've researched various options to implement caching at different layers. When it was clear that everything had a query parameter and that it could not be solved on the network layer, I looked at bottlenecks in their monolith through static code analysis. After founding two code bugs, I managed to get their monolith working by dockerizing it. |
| activities | Creating diagrams of current architecture Dataflow analysis Static code analysis (Ruby) Dockerizing monolith |
| techniques | Scrum draw.io LaTeX Ruby Nginx HAproxy CDN Content Delivery Network Javascript REST API HAL JSON Git |

|  |  |
| --- | --- |
| role | DevOps | imgZine (IT), Amsterdam |
| timespan | September 2015 -> January 2016, 40h p/w (internship) |
| desc | Developing a Proof of Concept backend which is used for chat inside intranet applications. Dockerized an existing XMPP server and compiled it with WebSocket libraries and developed an HTML5 app which connects to the backend using WebSocket (RFC7395). |
| activities | Developing HTML5 chat client Configure backend Installation script in Bash |
| techniques | Scrum Docker HTML5 XMPP server WebSocket Bash LaTeX Git |

|  |  |
| --- | --- |
| role | NodeJS developer | Maxedy (IT), Amsterdam |
| timespan | February 2015 -> June 2015, 40h p/w (internship) |
| desc | Deployment using Bash and REST-API in Meteor using OrientDB |
| techniques | Scrum OrientDB Node.js Javascript REST API Bash NoSQL HTML5 draw.io Git Mocha |

|  |  |
| --- | --- |
| role | Grails developer | Iceberg Webshophands (retail), Dronten |
| timespan | November 2013 -> January 2014, 40h p/w (internship) |
| desc | Proof of Concept; developing online craigslist for fresh food |
| techniques | Scrum Bash Java HTML5 SQL Grails Git |

|  |  |
| --- | --- |
| role | PHP developer | Windesheim (education), Zwolle |
| timespan | September 2011 -> December 2011, 30h p/w (edu. project) |
| desc | Development of two websites in teams |
| techniques | MySQL PHP5 svn |