What I need to know:

* Summary of the application, what it can do.
* **What are the user stories you want to address?**
  + Example: as a developer I want to be able to push directly to staging instances
* How many developers are there? How many are UI people? How many are dedicated Ops?
* Complete technology Stack *(How many Tiers does your application have?)*
  + - Database Questions:
      * how is connection pooling done?
      * What ORM are you using?
      * How is the transaction handled?
      * AWS Instance Type
      * What is the average IO of the application
      * Is there an existing load test benchmark? What tools were used?
      * Is the database encrypted? Is the Database connection secured?
      * How is the backup process?
    - Network
      * How many people have access to Prod environment, who are these people?
      * How many people have access to Prod Database
      * Are you able to trace who is accessing prod environments at any given time?
      * What are the ports open for your instances?
      * Is there an existing blacklist/whitelist for accessing each instance?
      * Is there any mechanism to block any DDOS attack?
      * Is there a bastion server setup to access prod environments, or do you directly SSH to them?
      * Where are the SSH keys stored? Who has access to them
    - Ops
      * How is the application packaged?
      * What is the build tool used?
      * How many environments are there? How many environments are used for testing/development purposes
      * How many manual steps are done before deployment to production is ready?
      * How is the deployment done to staging servers?
      * How are the sensitive details stored? example: database password, Are they different in each environment? How is the packaging resolving to the correct instance? Is there a separate build for prod/staging environments? How are different environments handled, ie Staging environments against Production environments?
      * How long does it take for a new developer (junior to mid-level) to set up a local environment? (seconds/hours/days)
      * How snapshots handled against release candidates?
      * How are deployment tracked? audit log of all previous deployment
      * How easily can I rollback from one version to an older version
      * What repository are you using? (Git, bitbucket, etc)
    - Application Layer
      * What are the technologies used? Front-end back-end, what versions? i.e Java 8, spring 3 etc
      * How is the application run, what is application server used?
      * How much part of the codes is covered in Regression Testing/Unit testing?
      * How many lines of codes? (estimate)
      * What is the code review process?
      * What are all the technology stack used in the application
      * Where are the bottleneck?
      * Is there penetration testing?
      * Is there load testing?
    - Documentation
      * Can you provide existing technical documentation?
      * How does anyone keep track of all the features made, is it tied to any release version?
      * How are the requirements given to the developers?
    - Others
      * What/How is the software development cycle?
      * Any form of caching
      * Do you have messaging?
      * Any other background processes?
      * What is the DR process?
      * Are there any customer sensitive data saved?
      * What is the process for PVT? How does the tester ensure that deployment was successful?
      * Are the customers able to access the application during deployment?
* Access to a running dev/local instance/code base
* Network Diagram (Example below)
* Data flow Diagram (Example below)



